

# Visualisierungsdesign für 3D-Benutzerschnittstellen unter Verwendung komponierter Darstellungsverfahren

Dissertation

zur Erlangung des akademischen Grades Doktoringenieur (Dr.-Ing.)  
vorgelegt an der Technischen Universität Dresden, Fakultät Informatik

eingereicht von

Dipl.-Medieninf. Jan Wojdziak

geboren am 01.05.1982 in Dresden

Betreuer und Gutachter:	Prof. Dr.-Ing. habil. Rainer Groh
Externer Gutachter:	Prof. Dr.-Ing. habil. Heidrun Schumann
Fachreferent:	Jun.-Prof. Dr.-Ing. Thomas Schlegel

Tag der Verteidigung: 24. Februar 2014

Dresden, September 2013





*Für Bine und Timo*





# Zusammenfassung

Das computergrafische Abbildungsverfahren zur Verwirklichung von dreidimensionalen Darstellungen ist ein wichtiges Instrument für die Gestaltung interaktiver 3D-Benutzerschnittstellen. Der Einsatz von Methoden und Techniken, die von der konventionellen Darstellungsweise der Zentral- und Parallelprojektion abweichen, bietet für die Verbildlichung von Daten und Informationen eine zusätzliche Vielfalt, Effektivität und Ausdruckskraft. Die Betrachtung von Projektionsverfahren abseits des bisher angestrebten Fotorealismus dokumentiert, dass durch nichtlineare und multiperspektivische Darstellungen spezifische Eigenschaften und Charakteristiken eines Datenbestandes vermittelt werden können. Infolgedessen kann durch ein erweitertes Repertoire an Abbildungstechniken die visuelle Kommunikation von 3D-Benutzerschnittstellen gezielt geführt und konstruktiv eingesetzt werden, um den bildgestützten Dialog zwischen Mensch und Computer zu bereichern. Indessen zeigt sich ein Missverhältnis zwischen den verfügbaren Abbildungstechniken im Bereich der computergrafischen Bilderzeugung einerseits und den Darstellungsweisen, die in praxisrelevanten und anwendungsbezogenen interaktiven 3D-Anwendungen eingesetzt werden, andererseits. Daraus resultiert zum einen die Fragestellung, in welcher Form beziehungsweise mit welchem Bestreben eine Integration der derzeit ungenutzten Abbildungstechniken in den interaktiven Nutzer-Bild-Dialog erfolgen kann. Zum anderen ist die Frage, durch welche Methoden und Vorgehensweisen eine Einbindung der Abbildungstechniken in den Entwurfsprozess von 3D-Interfaces – unter der Berücksichtigung des Anwenders und dessen konkreter Daten – unterstützt werden kann, bisher weitgehend unbeantwortet.

Auf Basis einer Analyse des computergrafischen Kameramodells zeigt die Arbeit auf, dass unterschiedliche Darstellungsweisen durch die Modifikation des Projektionsvorganges erreicht werden können. Darauf aufbauend wird die Spannweite existierender Abbildungstechniken systematisch auf deren Funktionsweise und deren Eigenschaften sowie Einsatzmöglichkeiten untersucht. Ferner wird verdeutlicht, dass für die Verwirklichung von Visualisierungszielen in interaktiven 3D-Benutzerschnittstellen, unter Einbeziehung der Abbildungstechniken, der Entwurfsprozess durch ein modellbasiertes Vorgehen methodisch und konzeptionell unterstützt und infolgedessen begünstigt werden kann. Ausgehend von herausgearbeiteten Gestaltungshinweisen für den Einsatz linearer und nichtlinearer Abbildungsverfahren erfolgt die Realisierung eines Vorgehensmodells zur Umsetzung von 3D-Benutzerschnittstellen. Die modellgestützte Verfahrensweise basiert zum einen auf der grafischen Modellierungssprache UML und zum anderen auf der deklarativen Auszeichnungssprache XML, um 3D-Benutzerschnittstellen zu entwerfen und darüber hinaus als maschinenlesbare Beschreibungen für interaktive Anwendungen bereitzustellen. Die Einbindung von Entwurfsmustern in die Vorgehensweise dient als ergänzendes Werkzeug für die Interfacemodellierung im Hinblick auf die Gebrauchstauglichkeit von 3D-Benutzerschnittstellen, da die Erfüllung software-ergonomischer Anforderungen in modellbasierten Entwurfsprozessen per se nicht berücksichtigt wird. Weiterhin wird für die Anwendung der vielfältigen Visualisierungsverfahren einerseits und die Überprüfung des Vorgehensmodells andererseits eine interaktive 3D-Anwendung vorgestellt. Auf dieser Basis werden Anwendungsszenarien dargelegt und erläutert, wodurch die vorgestellte Vorgehensweise zur Integration der bisher in interaktiven 3D-Interfaces ungenutzten Gestaltungsmöglichkeiten veranschaulicht werden kann.



# Vorwort

Die vorliegende Arbeit entstand an der Professur für Mediengestaltung der Technischen Universität Dresden, vor allem während meiner Mitarbeitertätigkeit im Förderprojekt der Landesinnovationspromotion zum Thema Visualisierungskonzepte und Interaktionsstrukturen im dreidimensionalen Datenraum (SAB-Nr.: 080937697). Das Projekt wurde durch die Europäische Union und den Freistaat Sachsen aus Mitteln des Europäischen Sozialfonds (ESF) gefördert.



## Danksagung

Ich danke Prof. Dr.-Ing. habil. Rainer Groh für die fachliche Betreuung und die Bereitstellung der Infrastruktur der Professur für Mediengestaltung, die wesentlich zum Gelingen der Arbeit beigetragen haben. Mein Dank schließt seine zielführende Unterstützung und konstruktive Kritik sowie seine Empfehlung zur Landesinnovationspromotion und die Freiheit bei der Themenwahl ein.

Für die bereitwillige Begutachtung dieser Arbeit möchte ich mich bei Prof. Dr.-Ing. habil. Heidrun Schumann und Jun.-Prof. Dr.-Ing. Thomas Schlegel bedanken. Die wertvollen Rat-schläge und die hilfreichen Anmerkungen im Zuge der Anfertigung der Dissertationsschrift haben mich motiviert und in der Fortführung bestärkt.

Mein Dank gilt darüber hinaus meinen Kollegen an der Professur für Mediengestaltung. Dabei danke ich insbesondere Ingmar S. Franke für die Motivation und die wichtigen Impulse während der gesamten Arbeit, Martin Zavesky und Dietrich Kammer für die Fachgespräche und die themenspezifischen Diskussionen sowie Christian Lambeck für die zahlreichen Gedankenaustausche. Schließlich möchte ich mich bei Franziska Hannß, Andreas Stahl, Romy Müller und Sandra Großmann sowie allen Mitarbeitern der Professur für Mediengestaltung für die hilfreiche gestalterische, technische und administrative Unterstützung bedanken. Weiterhin möchte ich meinen zahlreichen Beleg- und Diplomstudenten für ihre unverzichtbare praktische Unterstützung bei der Implementierung danken.

Einen besonderen Dank richte ich an meine Familie und meine Freunde für die vermittelte Zuversicht und für ihre nie nachlassende Unterstützung.

Dresden, im September 2013

## Hinweise zum Lesen des Dokuments

Einige der in dieser Arbeit genannten Firmen- und Produktnamen sowie Hardware- und Softwarebezeichnungen sind eingetragene Warenzeichen. Diese werden ohne Gewährleistung der freien Verwendbarkeit genutzt und sind im Text nicht gesondert ausgewiesen.

Aus Gründen der Lesbarkeit, sprachlichen Kürze und Ausgestaltung wird auf die explizite und zusätzliche Nennung der weiblichen Form bei Wörtern, beispielsweise Anwender und Entwickler, verzichtet. Die verwendete Form schließt jedoch alle Personen gleichberechtigt ein.

Sämtliche Abbildungen und schematische Darstellungen sind, wenn nicht explizit herausgestellt, vom Autor angefertigt. Bei Bildschirmschnappschüssen ist die Herkunft der gezeigten Anwendung im Text und in der Bildunterschrift ausgewiesen.

## Hinweise zum Satz und zur Formatierung

Zur besseren Verständlichkeit der vorliegenden Arbeit werden typografische und formative Kennzeichnungen eingesetzt. Diese dienen der Hervorhebung, Unterscheidung und Kennzeichnung inhaltlicher Aspekte. Hierzu werden Fachausdrücke, insbesondere bei erstmaliger Verwendung beziehungsweise deren Einführung, *kursiv* gekennzeichnet. Ferner werden direkte Zitate durch deutsche Anführungszeichen „“ kenntlich gemacht. Weiterhin sind durch den Autor geprägte Bezeichnungen und Begriffe durch französische Anführungszeichen »« hervorgehoben.

# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung.....</b>	<b>1</b>
1.1	Hintergrund und Motivation .....	1
1.2	Problemdarlegung und Zielsetzung.....	5
1.3	Aufbau der Arbeit.....	9
<b>2</b>	<b>Grundlagen interaktiver 3D-Anwendungen .....</b>	<b>11</b>
2.1	Einführung in computergrafische Darstellungen .....	12
2.1.1	Grundlagen und Begriffe .....	12
2.1.2	Ausprägungen computergrafischer Darstellungen .....	13
2.2	Charakteristik von 3D-Benutzerschnittstellen .....	15
2.2.1	Datenbild, Dialogbild und Interaktionsbild .....	16
2.2.2	Der Nutzer-Bild-Dialog.....	16
2.2.3	3D-Darstellungen in interaktiven Systemen .....	18
2.3	Entwicklung von 3D-Benutzerschnittstellen für interaktive Anwendungen .....	20
2.3.1	Modellgestützte Benutzerschnittstellenentwicklung.....	21
2.3.2	Entwurfsmuster im Interfacedesign.....	24
2.4	Spannungsfeld der Technischen Visualistik .....	25
<b>3</b>	<b>Abbildungsverfahren für interaktive 3D-Anwendungen.....</b>	<b>27</b>
3.1	Computergrafische Benutzerschnittstellen .....	28
3.1.1	Historischer Abriss .....	28
3.1.2	Anwendungsdomänen .....	30
3.1.3	Beispiele für interaktive 3D-Benutzerschnittstellen .....	34
3.1.4	Nutzungskontext von 3D-Benutzerschnittstellen .....	38
3.1.5	Notwendigkeit einer systematischen Betrachtung .....	40
3.2	Formen und Eigenschaften von 3D-Darstellungen.....	42
3.2.1	Lineare Darstellungsformen .....	45
3.2.2	Nichtlineare Darstellungsformen .....	46
3.2.3	Lineare und nichtlineare Darstellungsformen in 3D-Anwendungen.....	53
3.3	Die computergrafische Bildsynthese.....	53
3.3.1	Der Abbildungsprozess als Grafikpipeline .....	54
3.3.2	Das computergrafische Kameramodell.....	57
3.3.3	Eine Systematisierung computergrafischer Projektionen.....	67
3.4	Gestaltungshinweise für 3D-Interfaces.....	69
3.4.1	Akzentuierung in der Darstellung .....	70
3.4.2	Einsatz linearer und nichtlinearer Darstellungsformen .....	71
3.4.3	Koordination multipler 3D-Darstellungen.....	72

3.4.4	Gewährleistung der Echtzeitfähigkeit.....	73
3.4.5	Der interaktive Prozess mit 3D-Benutzerschnittstellen .....	73
3.4.6	Interdisziplinarität im Entwicklungsprozess interaktiver 3D-Anwendungen.....	74
3.4.7	Konklusion der Gestaltungshinweise.....	75
3.5	Systematisierung von 3D-Abbildungsverfahren .....	76
3.5.1	Eine aufgabenorientierte Systematisierung.....	77
3.5.2	Abbildungsverfahren für Überblicksdarstellungen .....	80
3.5.3	Abbildungsverfahren für den Zoom .....	81
3.5.4	Abbildungsverfahren für das Filtern .....	83
3.5.5	Abbildungsverfahren für Detaildarstellungen .....	85
3.5.6	Abbildungsverfahren für den Vergleich .....	87
3.5.7	Abbildungsverfahren für Verlaufsdarstellungen.....	89
3.6	Diskussion und Motivation.....	91
<b>4</b>	<b>BiLL: Ein Framework für den 3D-Interfaceentwurf .....</b>	<b>94</b>
4.1	Anforderungen und allgemeine Vorgehensweise .....	95
4.1.1	Ansprüche und Herausforderungen .....	95
4.1.2	Generelle Anforderungen .....	96
4.1.3	Anforderungen an Visualisierungsbausteine.....	96
4.1.4	Anforderungen an eine Arbeitsumgebung.....	97
4.2	Basistechnologien des Bildsprache LiveLab .....	98
4.2.1	Komponentensystem .....	99
4.2.2	Grafik-Engine.....	102
4.2.3	Benutzeroberflächen-Toolkit.....	104
4.2.4	Technologieauswahl für die Arbeitsumgebung .....	105
4.3	Die Architektur des Bildsprache LiveLab .....	106
4.3.1	Abbildungsverfahren als Softwarekomponenten.....	106
4.3.2	Aufbau und Funktionsweise der Softwarearchitektur.....	108
4.3.3	Komponentenkombination zur Bildraummodifikation .....	110
4.3.4	Entwicklungsstufen der Arbeitsumgebung.....	113
4.4	Zusammenfassung und Diskussion.....	114
<b>5</b>	<b>MosaIK: Modellbasierte 3D-Interfacekomposition .....</b>	<b>116</b>
5.1	Entwurfsverfahren und deren Eignung für 3D-Benutzerschnittstellen.....	117
5.1.1	Herausforderungen im Entwurfsprozess.....	117
5.1.2	System- und benutzerzentrierte Vorgehensweisen .....	119
5.1.3	Motivation für einen modellbasierten und mustergestützten Entwurfsprozess.....	122
5.2	Metamodelle für den 3D-Interfaceentwurf.....	124
5.2.1	UML als Modellierungssprache.....	125
5.2.2	PLML als Musterdefinitionssprache.....	126
5.2.3	XML als Beschreibungssprache.....	127
5.3	Das MosaIK-Vorgehen für den Interfaceentwurf .....	128

5.4	Die Spezifikationsphase .....	132
5.5	Die Modellierungsphase .....	133
5.5.1	Das abstrakte Benutzerschnittstellendesign .....	134
5.5.2	Entwurfsmuster: Dialoggestalter .....	135
5.5.3	Entwurfsmuster: Regisseur .....	138
5.5.4	Das konkrete Benutzerschnittstellendesign .....	140
5.5.5	Entwurfsmuster: Sichtweise .....	142
5.5.6	Entwurfsmuster: Panelraster .....	144
5.5.7	Entwurfsmuster: Wandler .....	147
5.5.8	Entwurfsmuster: Modifikation .....	150
5.6	Die Implementierungsphase .....	153
5.6.1	Konzeption der Beschreibungssprache .....	154
5.6.2	Knotendefinition für das Dialogbild .....	155
5.6.3	Trennung von Daten- und Dialogbild .....	156
5.6.4	Struktur und Veränderlichkeit des Mosalk-Graphs .....	156
5.6.5	Mosalk-Graph-Knotenmenge .....	159
5.6.6	Das finale Benutzerschnittstellendesign .....	167
5.7	Resultate und Zwischenfazit .....	168
<b>6</b>	<b>Mosalk und BiLL in der Anwendung .....</b>	<b>170</b>
6.1	Das Bildsprache LiveLab im Überblick .....	171
6.1.1	Charakteristik der Arbeitsumgebung .....	171
6.1.2	Der Funktionsumfang .....	171
6.1.3	Bestandteile der Arbeitsumgebung .....	172
6.2	Fallbeispiel: Produktvisualisierung .....	175
6.3	Anwendungsszenarien im Kontext wissenschaftlicher Visualisierungen .....	181
6.3.1	Anwendungsszenario: Unfallforschung .....	182
6.3.2	Anwendungsszenario: Systembiologie .....	185
6.4	Fazit aus den Betrachtungen .....	186
<b>7</b>	<b>Schlussbetrachtung .....</b>	<b>188</b>
7.1	Zusammenfassung .....	188
7.2	Fazit .....	191
7.3	Zukünftige Arbeiten .....	194
7.4	Abschließende Bemerkung .....	199
	<b>Anhang .....</b>	<b>200</b>
	<b>Abkürzungsverzeichnis .....</b>	<b>210</b>
	<b>Webreferenzen .....</b>	<b>211</b>
	<b>Literaturverzeichnis .....</b>	<b>213</b>
	<b>Verzeichnis eigener Publikationen .....</b>	<b>230</b>





# Abbildungsverzeichnis

Abbildung 1: 3D-Benutzerschnittstelle mit einer visuell-räumlichen Struktur.....	15
Abbildung 2: Modell des interaktiven Nutzer-Bild-Dialogs .....	17
Abbildung 3: Benutzerschnittstellen interaktiver 3D-Anwendungen .....	32
Abbildung 4: 3D-Benutzerschnittstellen für wissenschaftliche Visualisierungen. ....	35
Abbildung 5: 3D-Interfaces im Bereich der Informationsvisualisierung .....	36
Abbildung 6: 3D-Benutzerschnittstellen für Architektur- und Landschaftsvisualisierungen. ....	37
Abbildung 7: Schematische Darstellung linearer Darstellungsformen.....	46
Abbildung 8: Objektraumbasierte Multiperspektive im Bild „Der Zinsgroschen“ .....	47
Abbildung 9: Objektraumbasierte Multiperspektive im Bild „Stilleben mit Obstkorb“ .....	49
Abbildung 10: Bildraumbasierte Multiperspektive in Bildern .....	50
Abbildung 11: Krummlinige Darstellung im Bild „Schlafzimmer in Arles“ .....	51
Abbildung 12: Krummlinige Darstellungsformen in Bildern .....	52
Abbildung 13: Modell der Grafikpipeline für die computergrafische Bildsynthese.....	55
Abbildung 14: Kameramodell zur Erzeugung computergrafischer Abbilder .....	58
Abbildung 15: Kameramodell zur Erzeugung einer objektraumbasierten Multiperspektive ..	59
Abbildung 16: Kameramodell zur Erzeugung einer krummlinigen Perspektive.....	61
Abbildung 17: Kameramodell zur Erzeugung einer bildraumbasierten Multiperspektive .....	63
Abbildung 18: Kameramodell zur Erzeugung multipler Sichten .....	64
Abbildung 19: Kameramodell zur Verlagerung des Hauptpunktes.....	66
Abbildung 20: Kameramodell zur Erzeugung invers-zentralprojektiver Darstellungen .....	67
Abbildung 21: Bestandteile der OSGi-Architektur als Schichtenmodell, .....	100
Abbildung 22: Mögliche Zustände von OSGi-Bundles innerhalb des OSGi-Lebenszyklus....	101
Abbildung 23: Softwarearchitektur des Bildsprache LiveLab als Komponentenmodell.....	109
Abbildung 24: Darstellungsformen im Bildsprache LiveLab .....	111
Abbildung 25: Sichtweisen auf die Interfacespezifikation von 3D-Anwendungen .....	130
Abbildung 26: Das MosalK-Vorgehen für den Entwurfsprozess von 3D-Interfaces.....	131
Abbildung 27: Das MosalK:Abstrakt-Profil für den abstrakten 3D-Interfaceentwurf.....	135
Abbildung 28: Das Entwurfsmuster Dialoggestalter (abstraktes Interfacedesign) .....	137
Abbildung 29: Das Entwurfsmuster Regisseur (abstraktes Interfacedesign) .....	139
Abbildung 30: Das MosalK:Konkret-Profil für den konkreten 3D-Interfaceentwurf .....	141
Abbildung 31: Das Entwurfsmuster Sichtweise (konkretes Interfacedesign) .....	142
Abbildung 32: Beispiele von Kameramodellen auf Basis des Entwurfsmusters Sichtweise ..	143
Abbildung 33: Das Entwurfsmuster Panelraster (konkretes Interfacedesign) .....	145
Abbildung 34: Beispiele für Sichtenanordnungen mithilfe der Mustervorlage Panelraster ..	146
Abbildung 35: Das Entwurfsmuster Wandler (konkretes Interfacedesign) .....	148
Abbildung 36: Beispiel für die Funktionsweise des Entwurfsmusters Wandler.....	149

Abbildung 37: Das Entwurfsmuster Modifikation (konkretes Interfacedesign).....	151
Abbildung 38: Beispiel einer Sichtenkopplung mithilfe des Entwurfsmusters Modifikation	152
Abbildung 39: Schematische Darstellung der Konzeption des Mosalk:Dialogs.....	154
Abbildung 40: Grundlegende Struktur des XML-Schemas Mosalk-Knoten .....	155
Abbildung 41: Struktureller Aufbau des Mosalk-Graphs.....	157
Abbildung 42: Schematische Darstellung des Rasterknotens und dessen Attribute .....	160
Abbildung 43: Schematische Darstellung des Panelknotens und dessen Attribute .....	161
Abbildung 44: Schematische Darstellung des Bildraumknotens und dessen Attribute .....	162
Abbildung 45: Schematische Darstellung des Kameraknotens und dessen Attribute.....	164
Abbildung 46: Schematische Darstellung des Modifikatorknotens und dessen Attribute ...	166
Abbildung 47: Das 3D-Interface des Bildsprache LiveLab .....	173
Abbildung 48: Die 2D-Benutzeroberfläche des Bildsprache LiveLab .....	174
Abbildung 49: Das abstrakte 3D-Interfacedesign für das Übersicht-und-Detail-Szenario ....	177
Abbildung 50: Das konkrete Interfacedesign für das Übersicht-und-Detail-Szenario .....	179
Abbildung 51: Das finale Interface des Übersicht-und-Detail-Szenarios.....	180
Abbildung 52: Wissenschaftliche Visualisierungen aus dem Bereich der Unfallforschung ...	183
Abbildung 53: 3D-Interface mit der Darstellung eines Unfalls.....	184
Abbildung 54: Darstellung der Leber auf zellulare Ebene und auf Gewebeebene .....	185
Abbildung 55: Modellhafte Nachbildung einer Molekülstruktur der Leber .....	186
Abbildung 56: Struktur des Mosalk:Dialog XML-Schema Mosalk-Knoten .....	209

## Tabellenverzeichnis

Tabelle 1:	Übersicht über die bildstrukturellen Eigenschaften von Bildräumen .....	69
Tabelle 2:	Kategorien der Systematisierung von Abbildungsverfahren .....	80
Tabelle 3:	Übersicht über die Abbildungsverfahren für Überblicksdarstellungen .....	82
Tabelle 4:	Übersicht über die Abbildungsverfahren für den Zoom .....	84
Tabelle 5:	Übersicht über die Abbildungsverfahren für das Filtern .....	86
Tabelle 6:	Übersicht über die Abbildungsverfahren für Detaildarstellungen .....	88
Tabelle 7:	Übersicht über die Abbildungsverfahren für den Vergleich .....	90
Tabelle 8:	Übersicht über die Abbildungsverfahren für Verlaufsdarstellungen .....	92
Tabelle 9:	Übersicht über die Attribute des Rasterknotens .....	160
Tabelle 10:	Übersicht über die Attribute des Panelknotens .....	161
Tabelle 11:	Übersicht über die Attribute des Bildraumknotens .....	162
Tabelle 12:	Übersicht über die Attribute des Kameraknotens .....	164
Tabelle 13:	Übersicht über die Attribute des Modifikatorknotens .....	166
Tabelle 14:	Legende zu den UML-Profilen .....	200
Tabelle 15:	Übersicht über das Mosaik:Abstrakt-Profil .....	201
Tabelle 16:	Übersicht über das Mosaik:Konkret-Profil .....	202
Tabelle 17:	Kurzdarstellung des Entwurfsmusters Dialoggestalter .....	203
Tabelle 18:	Kurzdarstellung des Entwurfsmusters Regisseur .....	204
Tabelle 19:	Kurzdarstellung des Entwurfsmusters Sichtweise .....	205
Tabelle 20:	Kurzdarstellung des Entwurfsmusters Panelraster .....	206
Tabelle 21:	Kurzdarstellung des Entwurfsmusters Wandler .....	207
Tabelle 22:	Kurzdarstellung des Entwurfsmusters Modifikation .....	208

## Verzeichnis von XML-Listings

Listing 1:	Struktureller Aufbau des Mosaik-Graphs als baumartige XML-Struktur .....	157
Listing 2:	Definition eines Modifikatorknotens .....	166
Listing 3:	Das Übersicht-und-Detail-Szenario als Mosaik-Graph .....	181



# 1 Einleitung

*“Computer graphics has long been defined as a quest to achieve photorealism. As it gets closer to this grail, the field realizes that there is more to images than realism alone. Non-photorealistic pictures can be more effective at conveying information, more expressive or more beautiful.”* [DURAND 2002A, S. 1]

## 1.1 Hintergrund und Motivation

In der heutigen Zeit bieten Computer mannigfaltige Möglichkeiten der Erstellung, Bearbeitung und visuellen Darbietung von Daten. Die computergrafischen Systeme bereichern durch ihre Vielgestaltigkeit den Kommunikationsprozess des Menschen mit dem Computer, infolgedessen grafische Darstellungen ein unersetzbares Werkzeug in den Bereichen Wirtschaft, Wissenschaft und Unterhaltung sind. Die Entwicklung grafischer Anwendungen korreliert mit einer raschen Progression computergrafischer Darstellungsweisen, die mit einem Umdenken hinsichtlich deren Rolle im Prozess der Informationsdarbietung gekoppelt ist. Aufgrund der Leistungsfähigkeit heutiger Computer dringen dreidimensionale Darstellungen in immer weitere Anwendungsbereiche vor und ergänzen beziehungsweise ersetzen die zweidimensionalen Visualisierungen. Die zunehmende Bedeutung dreidimensionaler Darstellungen, die von BERTOLINE als *„renaissance in graphics“*<sup>1</sup> [BERTOLINE 1998, S. 181] postuliert wird, manifestiert sich dabei nicht nur in den Bereichen filmischer Spezialeffekte und Computerspiele, sondern ebenso in den Naturwissenschaften und im Ingenieurwesen.

Die Erzeugung einer 3D-Visualisierung wird aufgrund der inhärenten Tiefendimension des zugrunde liegenden dreidimensionalen *Objektraums* mithilfe eines *Projektionsvorgangs* auf einem flächigen Ausgabemedium abgebildet. Bei diesem computergrafischen Überführungsvorgang zeigen sich Parallelen zu den Verfahrensweisen des realen Fotografierens und des Filmens. Die *räumliche Struktur* einer *virtuellen Szene* entspricht hinsichtlich wesentlicher Merkmale einer realen Umgebung. Der Objektraum wird, ebenso wie es im optischen Verfahren der Lichtbilderstellung vollzogen wird, aus einer definierten *Perspektive* auf eine planare Fläche projiziert. Die Analogie zur Fotografie eröffnet für die Computergrafik die Möglichkeit, den Abbildungsprozess nach dem Vorbild des Fotoapparates effizient zu realisieren. Dazu ist die *computergrafische Kamera* im Aufbau und in der Funktionsweise einer Lochkamera nachempfunden. Der Abbildungsvorgang beruht auf mathematischen Gesetzen, die eine fotooptische Projektion idealisiert beschreiben. Das fotografische Abbild wird in der Folge partiell zum Vorbild für die computergrafische Abbilderzeugung. Aus dem dreidimensionalen Objektraum wird ein „fotorealistischer“ zweidimensionaler *Bildraum* generiert, infolgedessen

---

<sup>1</sup> „Renaissance im Bereich grafischer Darstellungen“

der Begriff des *Fotorealismus* „als Synonym für die bildtheoretische Stilcharakterisierung ‚realistisch‘ oder genauer eigentlich ‚naturalistisch‘ verwendet wird“ [SCHIRRA & SCHOLZ 2000, S. 69].

Zugleich eröffnen die Bestrebungen nach einer fotorealistischen Computergrafik den Blick auf die Restriktionen einer fotogleichen Darstellungsweise. Die exakte Repräsentation der Realität ist eine Einschränkung der grafischen Ausdrucksmöglichkeiten und darf nicht die alleinige Bestrebung bei der Bilderzeugung sein. Vielmehr sollte die Aussagekraft das entscheidende Kriterium für die subjektive Qualität eines Bildes sein (vgl. [WINKENBACH & SALESIN 1994, S. 91; DURAND 2002A, S. 116]) und das Verfahren im Abbildungsprozess bestimmen. Die dreidimensionale Computergrafik besitzt ein inhärentes Potenzial, die unterschiedlichsten Anforderungen an die Bildgestaltung zu erfüllen. Die Herausforderung, die sich mit dem in der operativen Nutzung computergrafischer Systeme bisher kaum genutzten Potenzial ergibt, liegt in der gezielten Anwendung von Darstellungsweisen zur Erzeugung grafischer Benutzerschnittstellen in der *Mensch-Maschine-Kommunikation* (vgl. [STROTHOTTE U. A. 1994, S. 455-456; DEUSSEN 2001, S. 79]). Der notwendige Verzicht auf den computergrafischen Fotorealismus als Visualisierungsziel entspräche nach den Ausführungen DEUSSENS einer Hinwendung zu „generalisierten grafischen Darstellungen“ [DEUSSEN 2001, S. 1], welche „die Vielfalt, Attraktivität und vor allem Effektivität“ [DEUSSEN 2001, S. 8] computergrafischer 3D-Benutzerschnittstellen steigern.

Das auf (formalisierten) Prinzipien beruhende Verfahren des computergrafischen Abbildungsschrittes bietet weder außerordentliche Vorteile für eine fotorealistische Darstellung noch ist es auf diese Darstellungsweise beschränkt. Ganz im Gegenteil: Das geometrische Modell der computergrafischen Kamera ist nutzbringend abstrakt. In dieser modellhaften Beschreibung ist der Projektionsschritt geradezu zweckmäßig die Erzeugung von Bildräumen für die Verwirklichung von Kommunikationsintentionen des Anwenders einzusetzen und die mannigfaltigen Darstellungsformen, einschließlich der naturalistischen Darstellung, in 3D-Benutzerschnittstellen zu integrieren. Dies gestattet dem Nutzer interaktiver 3D-Anwendungen situative und individuelle Einblicke in den Objektraum zu erhalten, Attribute des Objektraums zu akzentuieren und die visuellen Auswirkungen einer Objektveränderung im Bildraum augenblicklich zu erfahren. Dadurch wird der Betrachter in die Lage versetzt, nicht nur zu sehen, sondern auch zu erkennen, zu verstehen und zu bewerten.

Ob jedoch mit einer grafischen Darstellung ein kommunikatives Ziel erreicht werden kann, ist durch eine Vielzahl von Einflussfaktoren, die überwiegend kontextabhängig sind, bedingt. BERTIN bringt durch die Aussage „A graphic is not drawn once and for all“<sup>2</sup> [BERTIN 1981, S. 16] einerseits die Intention eine grafische Darstellung zielgerichtet darzubieten, andererseits die Mannigfaltigkeit der Ausdrucksmöglichkeiten, die keine geschlossene Lösung zulässt, zum Ausdruck. Wenn die Anforderungen, die an eine grafische Benutzerschnittstelle gestellt werden, durch eine Visualisierung nicht erfüllt werden können, wird deren kommunikatives Ziel nicht adäquat erreicht [SCHUMANN & MÜLLER 2000, S. 5 ff.]. Die Folge ist eine nur unzureichend angepasste Verbildlichung, die nach AMAR und STASKO zu einer Einschränkung

<sup>2</sup> „Eine Grafik ist nicht ein für allemal gezeichnet“

der Aufgabenunterstützung durch die grafische Schnittstelle führt: „*successful decision-making and analysis are more a matter of serendipity and user experience than of intentional design*“<sup>3</sup> [AMAR & STASKO 2004, S. 143].

Der Einsatz generalisierter computergrafischer Darstellungen in 3D-Benutzerschnittstellen ist jedoch, aufgrund einer ungenügenden systematischen Aufarbeitung der Darstellungstechniken einerseits und der fehlenden methodischen Unterstützung des Entwurfsprozesses andererseits, bisher kaum möglich. In der Folge werden die Möglichkeiten nicht-fotorealistischer computergrafischer Darstellungen – aufgrund mangelnder Kenntnis oder Vertrautheit – im Gestaltungs- und Entwurfsprozess interaktiver 3D-Anwendungen nicht eingesetzt (vgl. [ROSEN & POPESCU 2011, S. 631]). Dies steht im deutlichen Gegensatz zum Realisierungsprozess von grafischen 2D-Anwendungen, der durch eine Vielzahl von Methoden und Systematisierungen zur Konzeption und Realisierung der Mensch-Maschine-Kommunikation konsolidiert ist. Im Entwurfsprozess von 2D-Benutzeroberflächen werden aufgrund dessen Gestaltungsregeln, Vorgehensmodelle und Richtlinien erfolgreich angewendet. Die Unterstützung bei der Entwicklung von 3D-Benutzerschnittstellen ist hingegen durch einen unzureichenden Konkretisierungsgrad der Vorgaben, Richtwerte und Anwendungsregeln eingeschränkt. Eine wesentliche Ursache liegt in den bisher lediglich deskriptiven Gestaltungshinweisen für eine Realisierung [SHNEIDERMAN 2003, S. 14–15] bei einer gleichzeitig höheren Komplexität der 3D-Benutzerschnittstellen begründet. Dabei lässt sich durch *modellbasierte Vorgehensweisen* nicht nur der Verwirklichungsprozess von 3D-Benutzerschnittstellen sublimieren, sondern (zumindest potenziell) auch die Gebrauchstauglichkeit von 3D-Benutzerschnittstellen erhöhen. Auf Basis eines Vorgehensmodells können die unterschiedlichen Darstellungsweisen, in Abhängigkeit von den Nutzungszielen, als singuläre Bildräume festgelegt oder aber in Form von koordinierten multiplen Sichten zu hybriden 3D-Benutzerschnittstellen verbunden werden.

Im Hinblick auf die Gewährleistung der softwareergonomischen Qualität von Benutzerschnittstellen bieten *Entwurfsmuster* eine zusätzliche Unterstützung für Interfacedesigner und Entwickler. Der aus der Architekturtheorie stammende Musteransatz geht auf den Architekten ALEXANDER zurück, der Entwurfsmuster mit dem Ziel entwickelte, die Realisierung von komplexen Architekturstrukturen methodisch zu unterstützen (vgl. [ALEXANDER U. A. 1977]). Inzwischen werden weiterführend Lösungsvorlagen für wiederkehrende Problemstellungen bei der Entwicklung objektorientierter Software<sup>4</sup> und ebenso für den Entwurf von Interaktionsmethoden für traditionelle Anwendungen<sup>5</sup> und Internetwebseiten<sup>6</sup> entwickelt und angewendet. Für den Entwurf von 3D-Benutzerschnittstellen existiert bisher keine vergleichbare Unterstützung, wenngleich diese im Hinblick auf die Gebrauchstauglichkeit von 3D-Benutzerschnittstellen einen modellbasierten Entwurfsprozess von *3D-Interfaces* begünstigt. Die bisher fehlende konzeptionelle und methodische Unterstützung hemmt aber nicht nur den Entwurfsprozess, sondern ebenso die Transformation eines konkreten 3D-Interface in eine maschinenlesbare Interfacebeschreibung, wodurch die Bereitstellung eines Entwurfes als

<sup>3</sup> „die erfolgreiche Entscheidungsfindung und die Analyse sind eher auf den Zufall und die Nutzererfahrung zurückzuführen als auf eine intentionale Gestaltung“

<sup>4</sup> siehe [GAMMA U. A. 1994]

<sup>5</sup> siehe [BORCHERS 2001; TIDWELL 2011]

<sup>6</sup> siehe [VAN WELIE & VAN DER VEER 2003]

grafisches Werkzeug der Mensch-Maschine-Kommunikation in einer interaktiven 3D-Anwendung erschwert wird. Während die räumliche Struktur des Objektraums häufig in einer baumartigen Graphenstruktur auf Basis von *Szenenbeschreibungssprachen* definiert ist, werden wesentliche Eigenschaften des Bildraums in der jeweiligen interaktiven Anwendung verankert. Infolgedessen werden die Daten und deren Verbildlichung weder syntaktisch noch semantisch miteinander verknüpft. Durch eine Verschränkung der visuell abstrahierten Daten, die von GROH als *Datenbild* (vgl. [GROH 2007, S. 169]) beschrieben werden, und einer Beschreibung des Okulars auf diesen Daten – dem »*Dialogbild*«<sup>7</sup> (*Navigationsbild* nach [GROH 2007, S. 169]) – wäre die Voraussetzung für eine anwendungsunabhängige 3D-Benutzerschnittstellenbeschreibung geschaffen. Somit können fotorealistische und nicht-fotorealistische computergrafische Darstellungsweisen kontextabhängig und kommunikationsunterstützend an die zu visualisierenden Daten gebunden und in einer 3D-Benutzerschnittstelle komponiert und montiert werden. Durch diese Verschränkung in Form einer Beschreibung des *Interaktionsbildes* (vgl. [GROH 2007, S. 169]) wird der Anwender eines interaktiven 3D-Systems darin unterstützt, mit den Daten in einen interaktiven Dialog zu treten. Das 3D-Interface wird zum „*Kommunikationsmittel zwischen den Systemelementen Realität, Nutzer und Computer*“ [GROH 2007, S. 169].

## Vision

Der Mensch ist in der Lage aus seiner Umgebung bedeutsame Informationen zu extrahieren, zu nutzen und Zusammenhänge zu begreifen. Bei der Visualisierung dreidimensionaler Szenen werden die Fähigkeiten der menschlichen visuellen Wahrnehmung und Kognition unterstützt, um die relevanten Charakteristiken von räumlich verorteten Daten intuitiv erfassbar zu machen. Dazu sind in der Mensch-Maschine-Kommunikation das explizite Handeln und das implizite Verhalten des Nutzers in den interaktiven Prozess mit dem System eingebunden. In der Folge ist die 3D-Benutzerschnittstelle auf den Nutzer adaptiert und die visuellen Kommunikationsprozesse situativ ausgerichtet. Eine grafische Benutzerschnittstelle, die aus adaptierbaren und wiederverwendbaren 3D-Interfacebausteinen montiert wird, bildet das systemseitige Korrelat zur Unterstützung des *Nutzer-Bild-Dialogs*. Diese Softwarekomponenten sind plattformunabhängig und bieten vielgestaltige 3D-Visualisierungsverfahren für einen flexiblen Einsatz in Benutzerschnittstellen interaktiver Anwendungen an.

Da allein technische Rahmenbedingungen eine anwender- und kontextadaptierte 3D-Benutzerschnittstelle nicht gewährleisten können, ermöglicht ein allgemeingültiges Vorgehensmodell die Spezifikation und den Entwurf von 3D-Benutzerschnittstellen. Den Ausgangspunkt bilden dabei Gestaltungshinweise für 3D-Interfaces und eine Systematisierung von Visualisierungen, auf dem ein modellbasierter Entwurfs- und Entwicklungsprozess aufgebaut ist. Die innerhalb des Vorgehens eingesetzten Modelle basieren auf Beschreibungssprachen für 3D-Interfaces, die – unter Rückgriff auf plattform- und anwendungsunabhängige Visualisierungsbausteine – eine werkzeuggestützte Entwicklung von 3D-Interfaces ermöglichen. Ein

<sup>7</sup> Obgleich, dass der Begriff des Navigationsbildes die Menge der „interaktionsförderlichen Zustandsveränderungen im Datenbild“ [GROH 2007, S. 169] umreißt, fokussiert der Begriff der Navigation in dessen Wortbedeutung das Orientieren. Da jedoch der interaktive Dialog zwischen Nutzer und System sowohl das Orientieren als auch das Operieren (vgl. [GROH 2008, S. 4]) umfasst, wird im Kontext der Arbeit der Begriff des Dialogbildes anstelle des Navigationsbildes geprägt, da dieser sowohl das Orientieren als auch das Operieren subsumiert.



Katalog von Entwurfsmustern versetzt darüber hinaus Designer, Entwickler und Domänenexperten in die Lage, die softwareergonomischen Anforderungen an interaktive Systeme in effektive und effiziente 3D-Benutzerschnittstellen zu überführen.

Durch die systematischen und methodischen Vorgaben einerseits sowie die softwaretechnologischen Gegebenheiten andererseits ist ein effektiver Entwurfsprozess von domänenspezifischen interaktiven 3D-Echtzeitanwendungen zur Realisierung einer effizienten Mensch-Computer-Interaktion, bezogen auf die durchzuführende Aufgabe des Anwenders und dessen visuelle Wahrnehmung, gegeben.

## 1.2 Problemdarlegung und Zielsetzung

Im vorangegangenen Abschnitt wurde das derzeit weitgehend ungenutzte Potenzial, das sich aus der vorhandenen computergrafischen Darstellungsvielfalt ergibt, umrissen und dessen Einbindung in den Entwurfsprozess interaktiver 3D-Anwendungen motiviert. Im Folgenden werden konkrete Probleme, die innerhalb des umrissenen Themenfeldes bestehen, dargestellt und Forschungsziele abgeleitet.

### Problembereiche

- Es existiert eine Vielzahl von Visualisierungsverfahren, die auf Modifikationen der computergrafischen *Rendering-Pipeline* und insbesondere dem Projektionsschritt basieren. Diese werden überwiegend im Forschungsumfeld konzipiert und umgesetzt und dienen vorrangig dem Nachweis einer algorithmischen Realisierbarkeit eines Verfahrens oder einer Methode. Es bestehen kaum Systematisierungsansätze und keine Standardisierungsbemühungen, um die mannigfaltigen Projektionsverfahren zu charakterisieren und deren Einsatzbereiche zu identifizieren.
- Häufig erfolgt eine Übertragung von etablierten Konzepten und Metaphern des 2D-Interfacedesigns in den Entwurfsprozess von 3D-Benutzerschnittstellen. Aufgrund der zusätzlichen Freiheitsgrade und der differenten Rahmenbedingungen kann eine direkte Überführung zu einer erhöhten kognitiven Belastung des Nutzers führen. Gleichzeitig wird ein effektiver Einsatz des Interface bezogen auf das Visualisierungsziel beziehungsweise die durchzuführende Aufgabe nicht erreicht.
- Während für den Entwurf von 2D-Benutzeroberflächen mit Ordnungs- und Repräsentationssystemen, wie Raster und Bemaßungen, spezielle Kulturtechniken in der Interfacegestaltung genutzt werden, sind 3D-Benutzerschnittstellen überwiegend durch Bestrebungen geprägt, eine fotooptische Darstellung zu imitieren. Techniken und Verfahrensweisen, die eine Komposition und Montage der grafischen Elemente in 3D-Benutzerschnittstellen erreichen, werden im Entwurfsprozess von 3D-Anwendungen bisher kaum eingesetzt.
- Der Entwicklungsaufwand für die Realisierung interaktiver 3D-Anwendungen ist vergleichsweise hoch. Insbesondere für die Bereitstellung unkonventioneller Visualisierungsverfahren existieren kaum Wiederverwendungskonzepte und Softwarebausteine.

Dies führt zu einem Mehraufwand durch sich wiederholende Entwicklungsabläufe. Gleichzeitig sind für die Bereitstellung von Darstellungsverfahren ausgereifte Programmierkenntnisse notwendig, da eine Benutzerschnittstelle ohne die Verwendung bereits bestehender Komponenten und ohne die Unterstützung durch angepasste Entwurfswerkzeuge implementiert werden muss.

- Nur selten erfolgen der Entwurf und die Entwicklung von interaktiven 3D-Anwendungen in einem interdisziplinären Umsetzungsprozess unter Beteiligung von Interfacedesignern, Softwareingenieuren und Domänenexperten. Dies ist dem Fehlen eines gemeinsamen Vorgehensmodells geschuldet, wodurch eine systematische Entwicklung eingeschränkt ist und ein methodisches Vorgehen im Interfacedesign nicht erreicht wird. Die resultierenden Mängel in der Modellierung und der Realisierung von 3D-Benutzerschnittstellen führen im Projektverlauf häufig zu Komplikationen hinsichtlich der Laufzeit, des Budgets und der Umsetzung.
- 3D-Formate und Szenenbeschreibungssprachen bieten für die Charakterisierung von 3D-Benutzerschnittstellen selten Definitionsumfänge, die über eine Festlegung von Szenenelementen und deren interaktiven Verhaltens im Objektraum hinausgehen. Damit wird dem Anwender das Datenbild – jedoch nicht das Dialogbild – in einer plattform- und anwendungsunabhängigen Beschreibungssprache zur Verfügung gestellt.

## Annahmen

Aus der Motivation und den dargelegten Defiziten lassen sich die folgenden Arbeitsthesen ableiten:

- Häufig werden die Benutzerschnittstellen derzeitiger interaktiver 3D-Anwendungen, ausgehend von der computergrafischen Kamera und deren Parameter im virtuellen Raum und nicht vom erzeugten Abbild und dem Betrachter, konzipiert. Durch eine Systematisierung von strukturellen Eigenschaften computergrafischer Bildräume auf Basis bestehender Abbildungsverfahren können Methoden und Darstellungsformen für die Verwirklichung von Visualisierungszielen abgeleitet werden.
- Die Komposition von Objektraumelementen im Bildraum sowie die Montage von Bildräumen in einer Benutzerschnittstelle bieten einen Gestaltungsspielraum zur Umsetzung eines Nutzer-Bild-Dialogs. Die erforderliche Flexibilität wird erreicht, indem durch eine variante Anordnung von Sichten und einen veränderlichen Aufbau von Bildräumen vielfältige Gestaltungsmöglichkeiten eröffnet werden, welche durch eine ausschließlich fotorealistische Darstellungsweise nicht gegeben sind.
- Der Nutzer-Bild-Dialog ist ein interaktiver Prozess zwischen dem Nutzer und dem System. Die Bereitstellung eines Bildraums, der die Anforderungen an einen situativen Anwendungskontext erfüllt, ist durch eine alleinig systemseitige computergrafische Bildsynthese nicht möglich. Vielmehr muss das System, in Form von steuerbaren Automatismen, der Mensch-Maschine-Kommunikation assistieren. Nur durch einen bilateralen Verwirklichungsprozess können situative Benutzerschnittstellen realisiert werden.

- Die stagnierenden Entwicklungen im Bereich interaktiver 3D-Benutzerschnittstellen sind nicht auf ein fehlendes Fortschreiten im Bereich der Hardware zurückzuführen. Die Ursachen sind primär die fehlende Unterstützung im Entwurfsprozess sowie die unzureichenden Wiederverwendungsmöglichkeiten der Interfaceelemente. Durch die Bereitstellung von Entwurfsmustern, die in einem methodischen Vorgehensmodell Anwendung finden, können die Entwicklungen im Bereich interaktiver 3D-Benutzerschnittstellen gefördert werden.
- Deklarative Auszeichnungssprachen bieten für die Beschreibung von Benutzerschnittstellen interaktiver 3D-Anwendungen einen alternativen Realisierungsansatz zur konventionellen Interfaceprogrammierung. Eine einheitliche Beschreibung von Daten- und Dialogbild kann eine Wiederverwendbarkeit und eine plattformunabhängige Verfügbarkeit von 3D-Interfacebeschreibungen gewährleisten. Ferner können aufgaben- und nutzeradaptierte Benutzerschnittstellen einem breiten Anwenderkreis zugänglich gemacht werden.

## Zielsetzung

Aufbauend auf den erläuterten Defiziten und den gefolgerten Arbeitsthesen ist das Ziel dieser Arbeit einen Beitrag zur Unterstützung des Entwurfsprozesses von 3D-Benutzerschnittstellen interaktiver Systeme, unter Verwendung komponierter linearer und nichtlinearer Projektionsverfahren, zu leisten. Dies begründet sich aus der Vorstellung, aufbauend auf einer Systematisierung von Darstellungsverfahren, wiederverwendbare und anpassbare Visualisierungsbausteine zur interdisziplinären Entwicklung von 3D-Interfaces bereitzustellen und in einen methodischen Entwurfsprozess einzubinden.<sup>8</sup> Hieraus ergeben sich die folgenden konkreten Ziele:

- Eine Systematisierung von Projektionsverfahren und deren Einordnung in einen Nutzungskontext als Beitrag zur grundlegenden Klassifikation und künftigen Standardisierung von Visualisierungsmethoden zur Darstellung räumlicher Strukturen auf planaren Ausgabegeräten
- Die Ausarbeitung von Gestaltungshinweisen für 3D-Benutzerschnittstellen, welche die Erzeugung kommunikationsunterstützender Visualisierungen, unter Berücksichtigung von Nutzungsanforderungen, fördern
- Die Bereitstellung einer interaktiven Arbeitsumgebung zur Erforschung von Visualisierungsverfahren für die bildgestützte Interaktion im Anwendungskontext der Gestaltung interaktiver 3D-Anwendungen
- Die Entwicklung einer modellbasierten Vorgehensweise für den Entwurf und die Beschreibung von Benutzerschnittstellen interaktiver 3D-Anwendungen basierend auf Modellierungs- und Auszeichnungssprachen
- Die Konzeption von Mustervorlagen für den Entwurf von 3D-Benutzerschnittstellen, die an das modellbasierte Vorgehen angelehnt ist und den Entwurfsprozess durch die Bereitstellung von Entwurfswissen vereinfacht

---

<sup>8</sup> siehe Erläuterungen im Abschnitt 1.1)

## Abgrenzung

Diese Arbeit ist im Schnittgebiet verschiedener Fachrichtungen der Informatik angesiedelt. Die *Computergrafik* ist durch die Bildsynthese und die resultierenden computergrafischen Abbilder von wesentlicher Bedeutung. Der Bilderzeugungsprozess wird hinsichtlich der Modifizierbarkeit des Projektionsschrittes, jedoch nicht in Bezug auf Performanceverbesserung im Rendering oder die Entwicklung von *Schattierern*<sup>9</sup> betrachtet.

Das Gebiet der *Interface-* und *Interaktionsgestaltung* zeigt, dass eine grafische Benutzerschnittstelle einerseits durch deren Darstellungsinhalt, andererseits durch die bildstrukturellen Charakteristiken auf den Nutzer wirkt. Ein Schwerpunkt der Arbeit ist hierbei die Konzeption von generellen Gestaltungshinweisen für den Entwurf und die Entwicklung interaktiver 3D-Benutzerschnittstellen. Durch die Darlegung von Anwendungsszenarien aus dem Bereich wissenschaftlicher Visualisierungen wird die kontextbezogene Gestaltung lediglich tangiert. Auch wenn die Gestaltung ein wichtiges Element für die effektive und effiziente Nutzung von 3D-Anwendungen darstellt, nimmt die konkrete Ausgestaltung von Interfaces nur einen kleinen Teil der Arbeit ein. Vielmehr werden bildstrukturelle Eigenschaften identifiziert und systematisiert und dadurch ein Möglichkeitsraum für die Gestaltung von 3D-Interfaces eröffnet.

Der Entwurfsprozess von Benutzerschnittstellen wird darüber hinaus durch die Erkenntnisse der *Kognitionspsychologie* im Bereich der visuellen menschlichen Wahrnehmung und Aufmerksamkeit beeinflusst. Der aktuelle Forschungsstand der angewandten Kognitionsforschung wird im Rahmen dieser Arbeit berücksichtigt und gegenüber dieser reflektiert, jedoch nicht durch eigene Beiträge erweitert.

Die Bereitstellung einer interaktiven 3D-Arbeitsumgebung und die prototypische Realisierung von Visualisierungsverfahren als Softwarekomponenten tangieren den Fachbereich der *Softwaretechnologie*. Die Anwendungsentwicklung dient jedoch ausschließlich der Umsetzung und Erforschung von Darstellungsverfahren und -methoden im Kontext eines forschungsnahen Interfacedesigns.

Der Fokus der Arbeit ist auf das Forschungsgebiet der *Technischen Visualistik*<sup>10</sup>, dessen Schwerpunkt die Gestaltung interaktiver Systeme darstellt, gerichtet, da eine modellbasierte Vorgehensweise zur Unterstützung des Entwurfsprozesses von Benutzerschnittstellen interaktiver 3D-Anwendungen konzipiert wird. Dabei werden weder lexikalische noch syntaktische Interaktionstechniken betrachtet (vgl. [Dix u. A. 2003, S. 308–309]), sondern der bildgestützte Interaktionsprozess mit 3D-Benutzerschnittstellen in den Mittelpunkt der Betrachtungen gestellt. Eine modellbasierte und eine deklarative Beschreibungsform sowie Entwurfsmuster werden hierbei für den Entwurfsprozess, unter Berücksichtigung von Visualisierungsverfahren und -zielen, thematisiert. Dabei umfassen die Ausführungen in dieser Arbeit multiple Bildräume, denen ein beständiger Objektraum zugrunde liegt. Infolgedessen werden weder Aspekte in Bezug auf zeitabhängige visuelle Strukturen noch die Zeitwahrnehmung im

<sup>9</sup> *shader* engl. Schattierer; Hardware- oder Software-Module, die Bildsynthese-Effekte in der 3D-Computergrafik implementieren.

<sup>10</sup> Das Fachgebiet der Technischen Visualistik ist eine anwendungsorientierte Forschungsrichtung zur Untersuchung der Abbildbarkeit situativer und interaktiver Relationen der Nutzer untereinander und zu ihren Interaktionsobjekten unter der Bedingung der Zeitlichkeit (vgl. [GROH 2011, S. 8]).

Nutzer-Bild-Dialog und deren Rolle bei der Informationsverarbeitung des Nutzers thematisiert. Ferner sind die Ausführungen auf die Realisierung von Benutzerschnittstellen für planare Ausgabegeräte fokussiert und schließen gleichzeitig die Betrachtung von nutzerumschließenden *virtuellen Umgebungen* (CAVE<sup>11</sup>) oder *Head-Mounted Displays*<sup>12</sup> (HMD) technologisch und kontextuell aus.

## 1.3 Aufbau der Arbeit

Aufbauend auf der formulierten Zielsetzung gliedert sich die Arbeit in zwei wesentliche Beiträge. Einerseits erfolgt eine Systematisierung von computergrafischen Abbildungsverfahren und die Überführung dieser in Visualisierungsbausteine für 3D-Benutzerschnittstellen. Andererseits wird die Entwicklung eines Vorgehensmodells für den Entwurfs- und Erstellungsprozess von 3D-Benutzerschnittstellen für interaktive Anwendungen unter Einbindung der vielgestaltigen computergrafischen Abbildungsverfahren vollzogen. Im Hinblick auf die Gebrauchstauglichkeit grafischer Benutzerschnittstellen werden hierbei Entwurfsmuster als unterstützendes Werkzeug für die Interfacegestaltung in das Vorgehensmodell integriert. Abschließend wird anhand von Beispielszenarien aufgezeigt, dass die vorgestellte Vorgehensweise eine Unterstützung für die konkrete und anwendungsbezogene Realisierung von 3D-Benutzerschnittstellen bietet.

### Gliederung

Das Kapitel 2 gibt einen Einblick in den Themenbereich der *computergrafischen Darstellungen*, indem zunächst essenzielle Grundlagen und Begriffe eingeführt und erläutert werden. Anhand einer schlaglichtartigen Betrachtung wird nachfolgend die Vielgestaltigkeit computergrafischer Visualisierungen verdeutlicht. Ferner werden *3D-Benutzerschnittstellen* charakterisiert und deren Bestandteile für einen bildgestützten Interaktionsprozess zwischen Anwender und System illustriert. Eine Betrachtung der Entwicklungsschritte für die Verwirklichung von 3D-Benutzerschnittstellen interaktiver Anwendungen und die Darlegung unterstützender methodischer Werkzeuge schließt das Kapitel ab.

Das Kapitel 3 vertieft die Thematik interaktiver 3D-Benutzerschnittstellen. Dazu werden Einsatzmöglichkeiten durch die Beschreibung von Anwendungsdomänen und –beispielen identifiziert. Weiterhin erfolgt eine Analyse von 3D-Darstellungen in Bezug auf deren Einsatzmöglichkeiten in interaktiven Anwendungen. Die sich anschließende Betrachtung der Rendering-Pipeline und im Besonderen der computergrafischen Kamera zeigen Modifikationsmöglichkeiten des *computergrafischen Abbildungsprozesses* auf. Darauf aufbauend werden *Gestaltungshinweise* für den Einsatz unterschiedlicher Darstellungsverfahren in singulären und multiplen Sichten sowie deren Eignung im Anwendungskontext formuliert. Den Schwerpunkt des Kapitels bildet die systematische Betrachtung von *Projektionsverfahren* und *Visualisierungszielen* im Kontext interaktiver 3D-Anwendungen.

<sup>11</sup> *Cave Automatic Virtual Environment* engl. Höhle mit automatisierter, virtueller Umwelt; Ein Raum in dem mithilfe von mehreren Projektionen von unterschiedlichen Seiten eine virtuelle Umgebung geschaffen wird.

<sup>12</sup> *Head-Mounted Display* engl. Kopfgebundenes Display; Ein Gerät, das am Kopf des Anwenders befestigt ist, beispielsweise eine Videobrille, und als grafisches Ausgabemedium dient.

Das Kapitel 4 stellt die 3D-Komponentenarchitektur »*Bildsprache LiveLab (BiLL)*« als Arbeits- und Experimentierplattform für die Umsetzung der im dritten Kapitel systematisierten Visualisierungsverfahren vor. Die Notwendigkeit einer exakten Parametrisierung, flexiblen Anpassung und einfachen Kombination von Visualisierungsverfahren in einer interaktiven Anwendung ist hierbei als Motivation für die Realisierung der Arbeitsumgebung zu sehen. Dazu wird eine Anforderungsanalyse vorgenommen und darüber hinaus Komponentensysteme aus der Softwaretechnik und Visualisierungsbibliotheken aus der Computergrafik betrachtet. Die identifizierten Softwaresysteme werden in Bezug auf die Anforderungen reflektiert und der komponentenbasierte Aufbau des Bildsprache LiveLab vorgestellt. Die Funktionsweise als Arbeitsumgebung wird anschließend erläutert, indem die unterschiedlichen Stufen der Entwicklung von 3D-Benutzerschnittstellen aufgezeigt werden.

Das Kapitel 5 beschreibt die Vorgehensweise »*MosaIK*« für die *modellbasierte 3D-Interfacekomposition*, der die Modellierungssprache UML und die Auszeichnungssprache XML zugrunde liegt. Mit dem Ziel einer Einbindung der systematisierten Darstellungsverfahren in den *3D-Benutzerschnittstellenentwurf* werden zunächst Möglichkeiten zur Unterstützung des Entwurfsprozesses analysiert. Darauf aufbauend erfolgt die Vorstellung und detaillierte Beschreibung des MosaIK-Vorgehens. Begleitend dienen Entwurfsmuster, die im Hinblick auf den Konkretisierungsgrad den Entwurfsphasen angepasst sind, als strukturierte Lösungsvorlagen für den Entwurf von Benutzerschnittstellen interaktiver 3D-Anwendungen.

Das Kapitel 6 verdeutlicht die Anwendbarkeit der Arbeitsergebnisse unter Einbeziehung der Arbeitsumgebung BiLL. Dies erfolgt durch die Vorstellung eines *Fallbeispiels* sowie durch die Darlegung von zwei *Anwendungsszenarien*. Dabei wird aufgezeigt, dass die im fünften Kapitel vorgestellte MosaIK-Vorgehensweise den Entwurf und die Verwirklichung von 3D-Benutzerschnittstellen im Kontext wissenschaftlicher Visualisierungen unterstützt und in der Folge für die Gestaltung interaktiver 3D-Anwendungen eingesetzt werden kann.

Das Kapitel 7 fasst die Arbeit zusammen und zeigt den Beitrag zum Forschungsgebiet auf. Ein Ausblick beschreibt Themen, welche die vorgestellten Resultate thematisch fortführen und zeigt ferner die Forschungsaufgaben, die sich aus diesen ergeben.

## 2 Grundlagen interaktiver 3D-Anwendungen

Für eine Heranführung an den Gegenstand der Arbeit werden in diesem Kapitel die Grundlagen interaktiver 3D-Anwendungen thematisiert. Dazu erfolgen zunächst eine allgemeine Charakterisierung von 3D-Visualisierungen und die Erläuterung von Begriffen, die mit der computergrafischen Abbilderzeugung einhergehen (2.1). Zum Spektrum von 3D-Darstellungen gehören neben fotogleichen Abbildern auch nicht-fotorealistische Darstellungen, die im Fokus der Arbeit stehen. Daher erfolgen Erläuterungen zu diesem speziellen Themenbereich computergrafischer Darstellungen und deren möglichen Ausprägungen. Ferner wird dabei der Gebrauch nicht-fotorealistischer Darstellungsweisen für eine bildgestützte Interaktion als berechtigt und begründet aufgezeigt. Die nachfolgende Einführung von 3D-Benutzerschnittstellen charakterisiert diese als visuelles Kommunikationsmittel zwischen Nutzer und System, aufbauend auf deren Eigenschaften und der Funktionsweise interaktiver Anwendungen (2.2). Die Ausführungen veranschaulichen darüber hinaus die Vielgestaltigkeit von 3D-Benutzerschnittstellen, wodurch die Herausforderungen im Hinblick auf die Entwicklung interaktiver 3D-Anwendungen vergegenwärtigt werden. Aktuelle Vorgehensweisen im Interfacedesign interaktiver 3D-Anwendungen werden daher in diesem Kapitel überblickshaft dargelegt (2.3). Einen Schwerpunkt bildet hierbei der modellgestützte Entwurfsprozess von Benutzerschnittstellen für interaktive Anwendungen. Hierbei werden Verfahren sowie grundlegende Modelle und Sprachen für den Entwurf von 3D-Benutzerschnittstellen erläutert. Der Fokus der Betrachtung liegt auf Vorgehensweisen, welche die Fachlichkeit von der Technologie trennen und formale Modelle für einen geführten Entwurfsprozess bereitstellen. Einen zweiten Schwerpunkt bildet die einführende Erläuterung von Entwurfsmustern im Kontext des Interfacedesigns. Mustervorlagen bieten die Möglichkeit existierendes Wissen über das Interfacedesign, zur Unterstützung des Entwurfsprozesses, in wiederverwendbarer und adaptierbarer Form bereitzustellen.

Auch wenn modellbasierte Vorgehensweisen und Entwurfsmuster zunächst thematisch nebeneinanderstehend erscheinen, zeigen aktuelle Bestrebungen, dass in Verbindung mit Entwurfsmustern eine zweckmäßige Nutzung von modellbasierten Vorgehensweisen für den Entwurf interaktiver Benutzerschnittstellen stattfindet. Das Kapitel schließt mit einer Betrachtung der Herausforderungen, die bei der Entwicklung interaktiver 3D-Benutzerschnittstellen unter Verwendung linearer und nichtlinearer Darstellungsweisen bestehen (2.4). Dabei werden Aufgabenfelder der beteiligten Personengruppen, interdisziplinäre Aspekte im Entwurfsprozess und die Notwendigkeit einer gemeinsamen Terminologie zum Austausch von Ideen, Meinungen und technischer sowie gestalterischer Rahmenbedingungen aufgezeigt.



## 2.1 Einführung in computergrafische Darstellungen

### 2.1.1 Grundlagen und Begriffe

Die visuelle Repräsentation von virtuellen räumlichen Strukturen steht im Mittelpunkt dieser Arbeit. Es ist daher wesentlich, zunächst die Charakteristiken bildlicher Darstellungen und die beschreibenden Begrifflichkeiten ebenso wie deren technische Erstellung einführend zu betrachten. Eine computergrafische Visualisierung bietet dem Betrachter eine bildliche Repräsentation von Daten und Informationen. Da visuelle Darbietungen von den meisten Menschen schnell und leicht verstanden werden können, dienen diese unter anderem dazu, Sachaussagen, Gegenstände, Prozesse aber auch Gefühle und Emotionen auszudrücken und zu kommunizieren. BRADEN und HORTIN beschreiben diese visuelle Lesefertigkeit des Menschen als eine Fähigkeit, Bilder zu verstehen und zu benutzen. Dies schließt die Fähigkeiten bildhaft zu denken, zu lernen und sich auszudrücken explizit ein (vgl. [BRADEN & HORTIN 1982]). Der Gebrauch von Bildräumen umreißt im Dialog mit einem computergrafischen System sowohl Aspekte der Erzeugung und Darbietung als auch der Veränderung. Zu den Resultaten computergrafischer Bildgenerierung gehören Visualisierungen, die durch Messgeräte erfasste Daten, mathematische Modelle oder durch den Anwender erzeugte Daten darbieten. Die erzeugten Bildräume können sowohl auf fotorealistischen als auch nicht-fotorealistischen Darstellungsverfahren basieren. Da die in der 3D-Computergrafik sehr präsente Visualisierungsform fotorealistischer Abbilder nur einen Teilbereich grafischer Repräsentationen darstellt, umreißt DURAND das Fachgebiet computergrafischer Darstellungen folgendermaßen: „*Computer depiction deals with all aspects of picture production, and in particular it is concerned with both rendering and interaction.*“<sup>13</sup> [DURAND 2002A, S. 112].

Der Bilderzeugungsprozess umfasst den Überführungsvorgang eines *dreidimensionalen Objektraums* in einen *zweidimensionalen Bildraum*. Dieser Vorgang und die damit verbundenen begrifflichen Festlegungen werden unter anderem in [PANOFSKY 1980, S. 125–126; WILLATS 1997, KAP. 2] bildwissenschaftlich, andererseits in [DURAND 2002A, S. 113–114; AKENINE-MÖLLER U. A. 2008, S. 15 ff.] computergrafisch weiterführend erläutert.

- Der Objektraum umfasst alle virtuellen Objekte, die in der Gesamtheit eine *dreidimensionale visuelle Struktur* in einer virtuellen *3D-Szene* bilden. Diese visuelle Struktur, die durch die geometrischen Eigenschaften der Objekte und die räumlichen Verhältnisse zwischen den Elementen der Struktur gekennzeichnet ist, wird als *Primärgeometrie* bezeichnet.
- Der Überführungsvorgang des Objektraumes in ein planares Abbild – die *Projektion* beziehungsweise der *Abbildungsvorgang* – erzeugt den Bildraum. Dies erfolgt im weitesten Sinne ausgehend vom Augpunkt eines Betrachters und dessen Sicht – die *Perspektive* – auf den Objektraum. Diese durch die Position des Augpunktes eingenommene Perspektive und die Verfahrensweise der Abbildung auf die Fläche legen die Rahmenbedingungen für den Bildraum fest.

<sup>13</sup> „Der Themenbereich der computergrafischen Darstellungen befasst sich mit allen Aspekten der Bilderzeugung, insbesondere mit der Bildsynthese und der Wechselwirkung zwischen Bild und Betrachter.“



- Der Bildraum beschreibt das planare Gegenstück zum Objektraum. Dessen struktureller Aufbau, der durch die dargestellten Objekte verbildlicht wird, bildet die *Sekundärgeometrie* und lässt sich als eine Menge bildstruktureller Eigenschaften beschreiben. Der in der Arbeit gebrauchte Terminus *nichtlinear* in Abgrenzung zu *linearen* Bildstrukturen bezieht sich nicht ausschließlich auf krummlinige Darstellungsweisen (vgl. [FOLEY 1995, S. 237]). Gleichsam sind Anomalien – im Sinne von Abweichungen – einer konventionellen linearprojektiven Darstellung inbegriffen.
- In Abhängigkeit vom Erzeugungsprozess wird bei einer grafischen Darstellung zwischen einem *Bild* und einem *Abbild* unterschieden. Während ein *Abbild* eine Reproduktion beziehungsweise eine Imitation einer 3D-Szene ist, bezeichnet das *Bild* eine mit schöpferischen Mitteln wiedergegebene Darstellung einer visuellen Struktur auf der Fläche. Im Rahmen dieser Arbeit wird infolge dieser Festlegung eine Darstellung basierend auf einer einfachen zentral- und parallelprojektiven Projektion als *Abbild* bezeichnet. Anderenfalls wird für die visuellen Resultate der Begriff des *Bildes* verwendet.

In der 3D-Computergrafik wird der Erstellungsprozess einer Visualisierung als *Bildsynthese* bezeichnet und in Kapitel 3 der Arbeit thematisch vertieft. Durch das *Rendern*<sup>14</sup> erzeugte Einzelbilder können in einem nachgelagerten Verarbeitungsschritt beispielsweise zu Animationssequenzen oder Filmen zusammengefügt werden. Eine weitere Ausprägung der Abbild-erzeugung ist das *Echtzeit-Rendering*. Dabei wird eine kontinuierliche Folge von Abbildern zur Laufzeit einer Anwendung berechnet und auf dem Ausgabegerät präsentiert. Dadurch kann die Bildfolge vom Nutzer als dynamischer Prozess wahrgenommen und interaktiv beeinflusst werden. Der Vorgang der Bildsynthese wird durch die *Grafikpipeline*<sup>15</sup> modellhaft als Abfolge von Verarbeitungsschritten des Grafiksystems, ausgehend von einer dreidimensionalen Szene hin zu einem zweidimensionalen *Abbild* auf dem Ausgabegerät, beschrieben. Die Entwicklungen in den Bereichen der Hardware und Software haben dazu geführt, dass die noch vor wenigen Jahren vergleichsweise starre Abfolge an Bildsyntheseschritten (Fixed-Function Pipeline)<sup>16</sup> innerhalb der Grafikpipeline fortwährend flexibilisiert wird. Durch die Integration von anpassungsfähigen Softwaremodulen (Schattierer) wird eine erweiterte Funktionalität bei gleichzeitig erhöhter Veränderlichkeit realisiert (Programmable Pipeline)<sup>17</sup>. Das Pipelineprinzip der Bildsynthese für computergrafische Ausgaben ist anwendungs- und plattformübergreifend, wodurch es sowohl der grafischen Ausgabe in Computerspielen und 3D-Desktop-Systemen als auch den Anwendungen mobiler Endgeräte zugrunde liegt.

## 2.1.2 Ausprägungen computergrafischer Darstellungen

Während mit einer Vielzahl von *Renderverfahren* das Ziel einer fotogleichen Darstellung verfolgt wird, ist diese Beschränkung bei nicht-fotorealistischen Rendertechniken aufgehoben (vgl. [GREEN U. A. 1999, KAP. 2]). Der Begriff *Non-Photorealistic Rendering*<sup>18</sup> (NPR) wurde in

<sup>14</sup> *Rendern* engl. Bildsynthese; bezeichnet in der Computergrafik die Erzeugung eines Abbildes aus den Rohdaten

<sup>15</sup> Der Begriff wird synonym zum Begriff *Rendering-Pipeline* verwendet

<sup>16</sup> *Fixed Function Pipeline* engl. Leitung mit festen Funktionen; Bildsynthese bei der die Funktionen innerhalb der Verarbeitungsschritte fest vorprogrammiert sind und nur durch weniger Parameter beeinflusst werden kann

<sup>17</sup> *Programmable Pipeline* engl. Programmierbare Leitung; Bildsynthese bei der die Verarbeitungsschritte programmierbar und dabei in den Stufen der Ausführung festgelegt sind

<sup>18</sup> *Non-Photorealistic Rendering*, engl. nicht-fotorealistische Bildsynthese

den 1990er Jahren durch die ACM SIGGRAPH geprägt. Der Begriff wird bis heute verwendet, auch wenn die Protagonisten den Fachbereich durch Begriffe wie *expressive*-, *artistic*- und *painterly rendering* weiterführend differenzieren. Der Negation des Begriffes folgend wird im Non-Photorealistic Rendering die Zielsetzung der fotorealistischen Bilderzeugung explizit vermieden. Eine genauere Gegenüberstellung der Eigenschaften dieser gegensätzlichen computergrafischen Darstellungsweisen wird in [GREEN U. A. 1999, KAP. 2] detailliert aufgezeigt. Es ist festzuhalten, dass zum einen die vollständige Überführung des Objektraums in ein Abbild von GREEN als Merkmal identifiziert wird, das in einer fotorealistischen Darstellung erfüllt sein muss. Zum anderen wird lediglich fotorealistischen Abbildern die Notwendigkeit einer präzisen Nachbildung attestiert. Ferner sieht GREEN in nicht-fotorealistischen Darstellungsweisen eine Form subjektiver Informationsdarbietung. Infolgedessen sind NPR-Verfahren, im Gegensatz zu einer fotogleichen Darstellung, durch künstlerische Techniken wie etwa der Strichzeichnung, der Aquarellmalerei oder der comichaften Verstärkung von Konturlinien inspiriert.

Die Einsatzmöglichkeiten nicht-fotorealistischer Bildsyntheseverfahren sind vielfältig: illustrative Strichzeichnungen sind für den Anwender im Bereich technischer Beschreibungen und Anleitungen oftmals verständlicher als realistische Repräsentationen. Ferner sind in medizinischen Anwendungen fotorealistische Darstellungen eher die Ausnahme. Darüber hinaus werden NPR-Verfahren in interaktiven Anwendungen eingesetzt, in denen das Anfertigen von Skizzen und schematische Darstellungen bisher ohne Computerunterstützung erfolgte. In diesen Bereichen, beispielsweise der Architektur, dienen CAD-Programme zum Erstellen von Zeichnungen, wobei durch das nicht-fotorealistische Verfahren der Strichzeichnung der Detailgrad einen Planungsstand auch unter Verwendung interaktiver 3D-Anwendungen widerspiegeln kann (vgl. [COCONU U. A. 2006, S. 27]). In Bezug auf künstlerische Techniken werden NPR-Verfahren eingesetzt, um Techniken aus der Malerei in die computergrafische Bilderzeugung zu übertragen. Hierzu werden Bilder analysiert [AGRAWALA U. A. 2000, S. 2–4] sowie reale Malgründe und Zeichenwerkzeuge simuliert, um beispielsweise die Farbe auf einer Leinwand oder die Tinte auf einem Papier mithilfe von Rendertechniken nachzuahmen (vgl. [MEIER 1996, S. 477–478]).

Neben den Verfahren, die perspektivisch abgebildete Szenen nicht-fotorealistisch darstellen, gibt es Techniken, die eine Veränderung der perspektivischen Struktur des Abbildes zum Ziel haben. Hierzu zählen nichtlineare und multiperspektivische Projektionen. Bei diesen Verfahren wird mit den fotooptischen Gesetzen gebrochen und infolgedessen Bildräume erzeugt, die mithilfe eines Fotoapparates nicht realisiert werden können. Beispiele hierfür sind die Erzeugung einer umfassenden Rundsicht oder die Vereinigung mehrerer Kamerastandpunkte in einem Bild. Anwendung finden diese Verfahren in wissenschaftlichen Visualisierungen, um beispielsweise Simultanansichten zur Analyse eines Objektes aus mehreren unterschiedlichen Blickwinkeln zu erzeugen [POPESCU U. A. 2009]. Ebenso werden die Rendertechniken zur Reduktion perspektivischer Verzerrungen in Weitwinkeldarstellungen (vgl. [ZORIN & BARR 1995]) oder für die konsistente Zusammenführung mehrerer Teilbilder zu einer Panoramaansicht genutzt [AGRAWALA U. A. 2006].

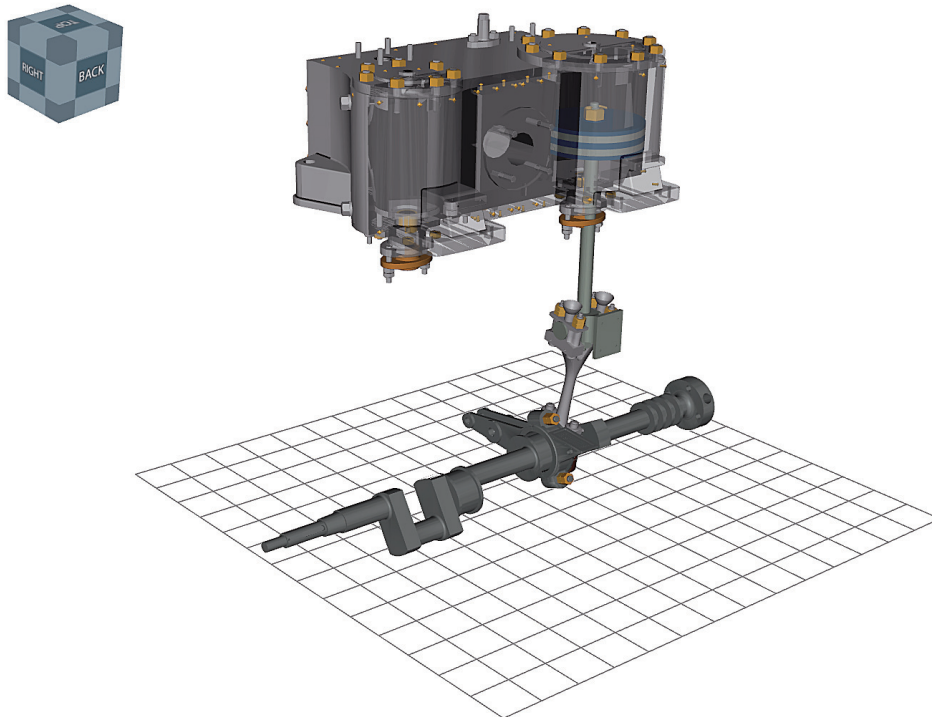


Abbildung 1: 3D-Benutzerschnittstelle mit einer visuell-räumlichen Struktur und einem Interaktionswerkzeug im linken oberen Bildbereich (Quelle: BiLL [@AG-TV])

## 2.2 Charakteristik von 3D-Benutzerschnittstellen

Die vorangegangene Einführung in computergrafische Darstellungen beschreibt bereits einige Eigenschaften von 3D-Benutzerschnittstellen (auch *3D-User Interface*, *3D-UI* und *3D-Interface*). Darauf aufbauend erfolgt nunmehr eine präzise Begriffsbestimmung und Charakterisierung. Eine 3D-Benutzerschnittstelle wird folgendermaßen definiert:

*»Eine 3D-Benutzerschnittstelle ist ein Softwarebestandteil eines Computersystems, die dem Benutzer den interaktiven, bildgestützten Dialog mit der Anwendung über eine oder mehrere projizierte computergrafische Darstellungen einer durch visuelle, räumliche Strukturen repräsentierten Datenmenge in Echtzeit ermöglicht.«*

Die Abbildung 1 zeigt exemplarisch eine 3D-Benutzerschnittstelle. Diese setzt sich aus einem interaktiven Abbild einer Datenmenge, als räumlich-verortete visuelle Struktur und zusätzlichen Interaktionsmöglichkeiten, die Zustandsveränderungen in der Benutzerschnittstelle bewirken, zusammen. Die visuelle Struktur einer 3D-Benutzerschnittstelle umfasst die Menge der virtuellen dreidimensionalen Objekte eines Objektraums, beispielsweise Produktmodelle – in Abbildung 1 der Teil eines Motors –, ebenso wie abstrakte Daten oder wissenschaftliche Messdaten.

## 2.2.1 Datenbild, Dialogbild und Interaktionsbild

Für die visuelle Präsentation von Daten auf einem Ausgabegerät werden diese zunächst akquiriert und aufbereitet, um die Datenmenge gegebenenfalls zu reduzieren und zu strukturieren. Die Resultate dieser Datentransformationen werden in geometrische Modelle überführt, indem in einer Daten-zu-Geometrie-Transformation die abstrakten Daten auf *visuelle Variablen* (*visuelle Attribute*) abgebildet werden. Bis zu diesem Punkt innerhalb des Verarbeitungsvorgangs kann die visuelle Struktur gewissermaßen als statische Repräsentation betrachtet werden. Anschließend erfolgt die grafische Anzeige auf dem Bildschirm oder einem anderen Medium, indem ein modifizierbarer Ausschnitt der Daten verbildlicht wird. Dadurch wird dem Anwender die Möglichkeit geboten, verschiedene Sichten auf die visuelle Struktur zu erzeugen, wodurch eine situative und nutzerabhängige Geometrie-zu-Bild-Transformation aus den geometrischen Strukturen ein Bild erzeugt. CARD u. A. beschreiben diese *Visualisierungspipeline* [CARD u. A. 1999, S. 17] als Referenzmodell der Visualisierung mit den Phasen *Datentransformation* (Vorverarbeitung), *Visuelle Abbildung* (Transformation in ein geometrisches Modell) und *Ansichtstransformation* beziehungsweise *Bildgenerierung*.

Der Prozess der Visualisierung sollte im Bezug auf die anwenderseitige Datensuche, -analyse und -präsentation zielgerichtet verlaufen. In der Folge wird eine auf die Daten und den Nutzer zugeschnittene Darstellungsweise benötigt. In Bezug auf die visuelle Datenverarbeitung in computergrafischen Benutzerschnittstellen wird in [GROH 2007, S. 169] die „mittels Computergrafik visualisierte Datenstruktur“ als *Datenbild* beschrieben, während der Schritt der Sichtstransformation „die interaktionsförderlichen Zustandsveränderungen im Datenbild“ [GROH 2007, S. 169] zeigt und in der gegenständlichen Arbeit als *Dialogbild* bezeichnet wird. Hierdurch werden dem Anwender bildgestützte Interaktionsmöglichkeiten basierend auf den zugrunde liegenden Daten geboten. Die 3D-Benutzerschnittstelle als visuelles Kommunikationsmittel zwischen dem Nutzer und dem System wird in der Zusammenführung von Datenbild und Dialogbild, basierend auf der Definition in [GROH 2007, S. 169] als *Interaktionsbild* bezeichnet.

## 2.2.2 Der Nutzer-Bild-Dialog

Interaktive 3D-Benutzerschnittstellen bieten einen bildgestützten Dialog zwischen dem Anwender und dem System an, mit dem Ziel die nutzerabhängige Geometrie-zu-Bild-Transformation zu vollziehen. Dabei wird der jeweilige Visualisierungsschritt systemseitig halb- beziehungsweise vollautomatisch ausgeführt, während der Nutzer diesen Vorgang kontrolliert und beeinflusst. Im Kontext der Mensch-Computer-Interaktion beschreibt Dix diese Kommunikation allgemeingültig als „communication between user and the system“<sup>19</sup> [Dix u. A. 2003, S. 124]. Damit weist Dix auf den dialogischen Charakter hin, ohne den ein grafisches System zu statischen Einzelabbildern beziehungsweise autonomen Animationen zurückgestuft wird. Yi greift die Definition auf und konkretisiert diese in Bezug auf die Kommunikation mit interaktiven Systemen als: „feature that provides users with the

<sup>19</sup> „Kommunikation zwischen Nutzer und dem System“

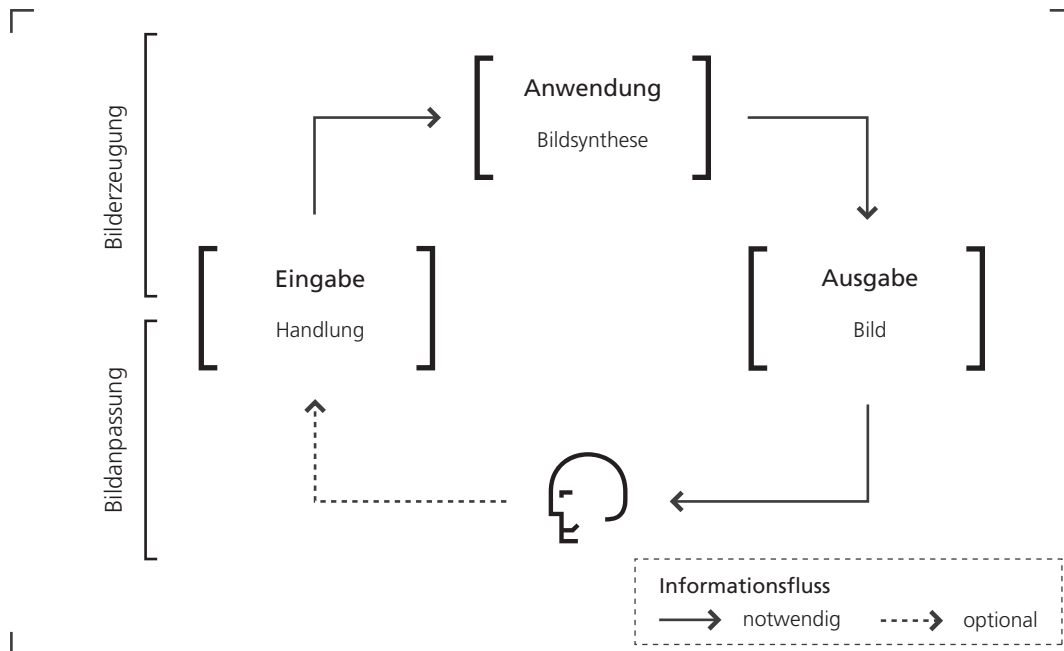


Abbildung 2: Modell des interaktiven Nutzer-Bild-Dialogs, das die Schritte der Bilderzeugung und der Bildanpassung als ein Zusammenwirken von Nutzer und Anwendung beschreibt.

*ability to directly or indirectly manipulate and interpret representations*<sup>20</sup> [YI U. A. 2007, S. 1225]. Yi stellt heraus, dass der Informationsfluss im Nutzer-Bild-Dialog im Wesentlichen vom System zum Nutzer erfolgt, aufgrund dessen im interaktiven Prozess die Anpassung und Veränderung von Repräsentationen vollzogen wird. In Bezug auf den computergrafischen Darstellungsprozess gestaltet sich die Kommunikation als eine Kombination aus der systemseitigen Sichterzeugung und der nutzerseitigen Anpassung des Interface. Die Sichttransformationen erfolgen durch eine Interaktion, bei der ein Anwender eine direkte Handlung mit der grafischen 3D-Benutzerschnittstelle vollzieht (vgl. [BOWMAN U. A. 2004, S. 7; DACHSELT 2004, S. 10–11]). Diese Wechselseitigkeit liegt darin begründet, dass unter Einbeziehung der grafischen Benutzerschnittstelle der Nutzer variierende Darstellungsziele verfolgt und die iterative Bildgenerierung einem Optimierungsprozess, bezüglich der Präferenzen des Anwenders, entspricht (vgl. [LANGE U. A. 2006, S. 121; LEVENE 1998, S. 12–13]). Dieses Vorgehen in Form eines interaktiven Prozesses erweitert die Unidirektionalität des Projektionsvorganges in interaktiven computergrafischen Anwendungen und erklärt gleichzeitig die Vielfalt bestehender Visualisierungsformen. Die schematische Darstellung in Abbildung 2 verdeutlicht den Dialogablauf, der als ein iterativer Vorgang bestehend aus Bilderzeugung und Bildanpassung gestaltet ist. Ausgehend von einer Anwendung erfolgt die Bildsynthese wie diese in Abschnitt 3.3 erläutert wird. Das erzeugte Abbild wird auf dem Ausgabegerät dargestellt und bildet die Schnittstelle zum Anwender. Diesem stehen durch den interaktiven Charakter der 3D-Benutzerschnittstelle nunmehr unterschiedliche Handlungsoptionen, zur Verfügung. Durch diese ist es dem Nutzer möglich, den Bilderzeugungsprozess zu verändern und eine modifizierte Bildsynthese zu initiieren.

<sup>20</sup> „die Funktionalität, die einen Nutzer dazu befähigt die dargestellten Repräsentationen direkt oder indirekt zu manipulieren und zu interpretieren.“

Es ist zu konstatieren, dass eine interaktive 3D-Anwendung nicht zwingend 3D-Interaktionen einschließen muss. Ein Nutzer kann Änderungen in einer interaktiven 3D-Anwendung ebenso über eine 2D-Benutzerschnittstelle vornehmen, indem dieser beispielsweise über ein 2D-Menü einen Betrachterstandpunkt im 3D-Objektraum festlegt. Oftmals wird der Nutzer-Bild-Dialog in interaktiven 3D-Anwendungen auch mittels spezialisierte Ausgabegeräte (CAVE, Head-Mounted Display) und Eingabegeräte (*Fly-Stick*<sup>21</sup>, *Motion Capture*<sup>22</sup>) vollzogen, welche die Art des Interaktionsprozesses beeinflussen. Der Fokus dieser Arbeit liegt jedoch auf dem Gebiet der bildgestützten Interaktion von 3D-Benutzerschnittstellen auf planen Ausgabegeräten, unter Einbeziehung etablierter Eingabe- und Ausgabegeräte. Für eine genauere Betrachtung von Eingabegeräten und Interaktionstechniken spezieller virtueller Umgebungen wird vielmehr auf ausführliche Erläuterungen und Systematisierungen, beispielsweise in [BOWMAN & HODGES 1997] für Eingabe- und Interaktionsformen in virtuellen Umgebungen, in [DACHSELT 2004] für Werkzeuge zur Interaktion in 3D-Umgebungen und in [QI U. A. 2006] für die räumliche Darstellung auf Head-Mounted Displays, verwiesen.

### 2.2.3 3D-Darstellungen in interaktiven Systemen

Eine interaktive 3D-Anwendung ist nach PIERCE [PIERCE 2001] ein Softwaresystem, das einem Nutzer eine dreidimensionale Szene darbietet und ferner ermöglicht, mit dieser virtuellen Struktur in *Echtzeit* zu interagieren. Durch diese allgemeine Definition wird ein großes Spektrum an computergrafischen Anwendungen berücksichtigt. Konkretisiert wird diese Beschreibung im Kontext der Arbeit daher durch den Fokus auf interaktive 3D-Anwendungen für konventionelle Desktop-Computersysteme. Während immersive 3D-Applikationen durch spezialisierte Hardware ein realistisches Eintauchen in teilweise benutzerumschließende virtuelle Umgebungen erlauben, gestatten 3D-Anwendungen mit einer Darstellung auf einem oder mehreren Computerbildschirmen die Verwirklichung von nicht-immersiven 3D-Applikationen. Die Anwendungen können ohne spezielle Ein- und Ausgabegeräte ausgeführt werden, wodurch diese für einen vielfältigen Einsatz prädestiniert sind. Gleichzeitig ist jedoch der Einsatz räumlicher Eingabegeräte, beispielsweise Fly-Stick, oder (auto)stereoskopischer Ausgabegeräte (3D-Monitore) nicht ausgeschlossen.

Die Bereitstellung anwendungsbezogener Daten sowie die Festlegung des strukturellen Aufbaus von Primärgeometrien erfolgt in den Anwendungen vorwiegend auf Basis von Austauschformaten für 3D-Szenen. Diese Szenenbeschreibungen können in einer Anwendung mithilfe interaktiver Werkzeuge erstellt und modifiziert oder durch Verfahren der Datenaufbereitung und des Datenmappings erzeugt werden. Eine zugrunde liegende *Szenenbeschreibungssprache*<sup>23</sup> ermöglicht die Charakterisierung der Szenenelemente in einer hierarchischen Struktur; einem *Szenengraph*. Ein Szenengraph ist ein gerichteter azyklischer Graph, der sich aus Knoten, Knotenverbindungen und Knotenkomponenten formiert und der hierarchischen Beschreibung einer dreidimensionalen Szene dient (vgl. [WALSH 2002, S. 18]). Neben proprietären Szenenbeschreibungssprachen, beispielsweise die Alias Studio Tools SDL,

<sup>21</sup> *Fly-Stick* engl. Steuerhebel, Joystick; Eingabegerät für Computer und Spielkonsolen, das dem Steuerknüppel eines Flugzeuges nachempfunden ist und zur Navigation in Verbindung mit optischen Trackingsystemen genutzt wird.

<sup>22</sup> *Motion Capture* engl. Bewegungserfassung; ist eine Technik zur Erfassung und Verarbeitung von menschlichen Bewegungen im Computer.

<sup>23</sup> Der Begriff wird synonym zum Fachausdruck Scene Description Language (SDL) verwendet



sind bestehende Sprachen vorwiegend XML-basiert, wobei *X3D* [WEB3D] und *COLLADA* [KHRONOS GROUP] zwei wesentliche Sprachen für die Beschreibung und den Austausch von 3D-Daten sind.

Die Verarbeitung der geometrischen Daten zur Darbietung auf dem Ausgabegerät wird häufig auf Basis einer imperativen Anwendungsprogrammierung unter Zuhilfenahme von Schnittstellen und Programmbibliotheken verwirklicht. Die am weitesten verbreiteten hardwarenahen (low-level) *Grafik-APIs*<sup>24</sup> für die Erzeugung von 3D-Computergrafik sind die plattformunabhängige Grafikbibliothek *OpenGL* und die Programmierschnittstelle *Direct3D* für die Entwicklung von Anwendungen auf Windows-Plattformen. Auf diesen hardwarenahen Schnittstellen bauen wiederum Bibliotheken auf, die zur Gruppe der high-level Schnittstellen gehören. Die vorwiegend objektorientierten *Toolkits*<sup>25</sup> bedienen sich dem Szenengraphprinzip zur Beschreibung der zu verarbeitenden Primärgeometrie, wodurch diese Werkzeuge eine Abstraktion zu den hardwarenahen Programmierschnittstellen bieten. Im Allgemeinen werden die Elemente des Objektraums als Transformationshierarchie strukturiert. Dies bietet die Möglichkeit Eigenschaften der Objektknoten an die untergeordneten Hierarchien zu übertragen, wodurch sich die Szenenelemente auf Basis des Szenengraphs strukturiert verarbeiten und das Bild als Ergebnis des Renderns auf dem Ausgabegerät darbieten lassen. Für die Verbildlichung des Objektraums wird der Szenengraph, traversiert, wobei die Informationen für die Bildsynthese erfasst werden und die notwendigen Aufrufe der hardwarenahen Grafik-API erfolgen. Verschiedene Grafikbibliotheken und 3D-Entwicklungswerkzeuge wurden in den letzten 20 Jahren entwickelt (vgl. [STEED 2008, S. 33]). Beispielsweise wurde *Iris Performer* [ROHLF & HELMAN 1994] für die Programmierung von 3D-Anwendungen zu Simulationszwecken und *Open Inventor* als erweiterbares und objektorientiertes Toolkit für 3D-Grafikanwendungen realisiert. (vgl. [WERNECKE 1994]). Neuere 3D-Toolkits, zum Beispiel *OpenSG* [REINERS U. A. 2002], *Juggler* [BIERBAUM U. A. 2001] und *ARToolkit* [KATO U. A. 2000] dienen in spezifischen Anwendungsdomänen als Werkzeug, wurden jedoch teilweise nur über einen begrenzten Zeitraum entwickelt und eingesetzt (vgl. [STEED 2008, S. 33–34; DACHSELT 2004, S. 9]). Gegenwärtig dienen Grafik-Toolkits wie beispielsweise *OpenSceneGraph* [WANG & QIAN 2010], aufgrund des gebotenen Funktionsumfangs und der Leistungsmerkmale, als Grundlage für 3D-Darstellungen interaktiver Anwendungen in vielen naturwissenschaftlichen und ingenieurtechnischen Bereichen. Weiterhin existieren durch den sich entwickelnden Spiele- und Unterhaltungsbereich spezifische Toolkits (*Grafik- oder Spiele-Engine*), beispielsweise *Irrlicht* [IRRLLICHT], *OGRE3D* [OGRE] und *Virtools* [DASSAULT SYSTEMES], die auf den low-level Grafik-APIs aufbauen und die szenengraphbasierten Beschreibungsformate unterstützen.

Die Verwirklichung des computergrafischen Bildraums in interaktiven Anwendungen ist indes sehr unterschiedlich. Wie durch die eben beschriebenen Toolkits deutlich wird, dient einerseits die imperative Programmierung, mithilfe existierender Programmiersprachen, für die Realisierung der Benutzerschnittstelle. Die bevorzugten Sprachen sind im Anwendungsfeld

<sup>24</sup> *Application Programming Interface*, engl. Schnittstelle zur Anwendungsprogrammierung; ist ein Teil eines Computerprogramms, der zur Anbindung anderer Programme an das Softwaresystem zur Verfügung gestellt wird

<sup>25</sup> *Toolkit* engl. Werkzeugsatz; bezeichnet eine Sammlung von Bibliotheken, Softwareklassen und Schnittstellen zur Reduzierung der Implementierungskomplexität und des Implementierungsumfangs von Programmen

computergrafischer Systeme C und C++. In der Umsetzung ergibt sich eine große Flexibilität im Entwurf von 3D-Benutzerschnittstellen bei gleichzeitig hohem Programmieraufwand, der lediglich durch die verfügbaren Grafik-Schnittstellen und Softwarebibliotheken sowie durch die grafische Unterstützung von *GUI-Buildern*<sup>26</sup> vermindert werden kann. Andererseits erfolgt zunehmend die Beschreibung des 3D-Interface mittels deklarativer Auszeichnungssprachen. Diese *Benutzerschnittstellenbeschreibungssprachen (UIDL*<sup>27</sup>) basieren überwiegend auf XML, deren Charakteristik in [SOUCHON & VANDERDONCKT 2003] folgendermaßen formuliert wird:

*“A UI Description Language (UIDL) consists of a high-level computer language for describing characteristics of interest of a UI with respect to the rest of an interactive application. Such a language involves defining a syntax [...] and semantics [...]. It can be considered as a common way to specify a UI independently of any target language (e.g., programming or markup) that would serve to implement this UI.”*<sup>28</sup> [SOUCHON & VANDERDONCKT 2003, S. 1]

Die Ausrichtung und die Ausprägung der Beschreibungssprachen sind abhängig von den Entwurfszielen, die mit der jeweiligen Sprache verwirklicht werden sollen. In der Folge orientieren sich die bestehenden Sprachen am Aufbau und der Funktionsweise der jeweiligen 3D-Anwendung. Beispielsweise dient die proprietäre Beschreibungssprache von Alias Systems [ALIAS SYSTEMS CORPORATION 2005] zur Beschreibung von Szenen in den 3D-Werkzeugen des Unternehmens oder die POV-Ray Szenenbeschreibungssprache [POV-RAY] zum Rendern von visuellen Strukturen. Wiederum andere Sprachen sind für eine definierte Zielplattform (vgl. [NAUJOKS 2008]) oder eine bestimmte Endgerätegruppe (vgl. [BIERMANN & JUNG 2004]) konzipiert. Den Sprachen ist gemein, dass diese im Vergleich zur Implementierung in einer Programmiersprache, unter Einbindung von Toolkits und GUI-Buildern, eine abstrakte Beschreibung von Bildräumen beziehungsweise Benutzerschnittstellen bieten.

## 2.3 Entwicklung von 3D-Benutzerschnittstellen für interaktive Anwendungen

Die Erforschung und Entwicklung grafischer Benutzerschnittstellen etablierte sich in den 1970er Jahren. Forscher des Xerox PARC<sup>29</sup> entwickelten im Jahr 1973 mit dem *Xerox Alto* einen Computer, der über eine grafische Benutzeroberfläche verfügte, welche die Desktop-Metapher umsetzte. Das Interface umfasste bereits die Elemente, die bis heute das Grundkonzept von 2D-Benutzeroberflächen bilden. In den frühen 1980er Jahren war die Konzeption und Realisierung von Benutzerschnittstellen mithilfe von *Bedienoberflächen Management Systemen (UIMS*<sup>30</sup>) ein Schwerpunkt im sich herausbildenden Forschungsbereich

<sup>26</sup> Ein GUI-Builder oder GUI-Designer ist ein Teil einer Entwicklungsumgebung, welcher die Entwicklung der grafischen Benutzerschnittstelle vereinfacht, indem das Anordnen von Steuerelementen auf Basis eines grafischen Interfaces erfolgt.

<sup>27</sup> *User Interface Description Language*, engl. Benutzerschnittstellenbeschreibungssprache

<sup>28</sup> „Eine Benutzerschnittstellenbeschreibungssprache besteht aus einer High-Level-Programmiersprache zur Beschreibung der Eigenschaften eines Interface mit Bezug auf die weiteren Bestandteile einer interaktiven Anwendung. Eine solche Sprache beinhaltet die Definition einer Syntax und einer Semantik. Es kann als eine Beschreibungsform zur Spezifikation eines Interface, unabhängig von der Zielsprache (beispielsweise Programmierung oder Markup), betrachtet werden, welche wiederum dazu dient die Benutzerschnittstelle zu implementieren.“

<sup>29</sup> Abkürzung für Palo Alto Research Center

<sup>30</sup> Abkürzung für User Interface Management System



der Software-Benutzerschnittstellen (vgl. [MYERS U. A. 2000, S. 10–11]). Ein UIMS ermöglicht die Beschreibung einer interaktiven Benutzerschnittstelle in einer maschinenunabhängigen, abstrakten Benutzerschnittstellenbeschreibungssprache, welche ferner die Ein- und Ausgabegeräte berücksichtigt. Eine Interfacebeschreibung wird auf Basis eines UIMS automatisch in ein ausführbares Programm übersetzt oder während der Laufzeit einer Anwendung interpretiert, um eine Standarddarstellung einer Programmoberfläche zu erzeugen. Das Ziel der Entwicklungen war nicht nur die vereinfachte Realisierung von Benutzeroberflächen, sondern auch die Verbesserung einer plattformübergreifenden Konsistenzwahrung sowie die Trennung von Benutzeroberfläche und Anwendungslogik.

Obwohl der UIMS-Ansatz ein flexibles und mächtiges Konzept für die praktische Anwendung darstellt, fand es zunächst keine nachhaltige Verbreitung und Nutzung (vgl. [MYERS U. A. 2000, S. 11]). Durch die zunehmende Variantenvielfalt von Endgeräten und die erweiterten Darstellungsmöglichkeiten sowie die Integration neuartiger Interaktions- und Visualisierungstechniken in die Mensch-Maschine-Kommunikation wird das Konzept der UIDL in den letzten Jahren verstärkt im Bereich ubiquitärer Systeme, multimodaler Anwendungen und interaktiver 3D-Anwendungen betrachtet und untersucht (vgl. [SHAER U. A. 2008; DACHSELT U. A. 2007]).

### 2.3.1 Modellgestützte Benutzerschnittstellenentwicklung

Der Entwicklungsprozess zur Verwirklichung von 3D-Anwendungen variiert in Abhängigkeit vom Anwendungskontext der zu erstellenden Software. Beispielsweise unterscheidet sich der Prozess für die Erstellung eines 3D-Computerspiels von einem Vorgehen zur Entwicklung einer interaktiven 3D-Produktpräsentation in vielen Details. Unabhängig vom konkreten Verwendungszweck stellt die Realisierung jedweder komplexer interaktiver 3D-Anwendung noch immer hohe Anforderungen an die Entwickler. Die 3D-Inhalte müssen spezifiziert und die visuelle Ausgabe sowie die Anwendungslogik programmiert werden, um schließlich alle Bestandteile in einem funktionierenden Gesamtsystem zusammenzuführen. Die Analyse, die Konzeption und die letztendliche Implementierung aller Softwarebestandteile als eine Gesamtanwendung erfordert mit einem steigenden Komplexitätsgrad einer interaktiven Anwendung einen strukturierten Benutzerschnittstellenentwurf, um die Umsetzung definierter Anforderungen an eine 3D-System unter Berücksichtigung der Projektlaufzeit sicherzustellen. Dabei sind formale Systeme, respektive künstliche Sprachen dienlich, da diese sowohl Anweisungen als auch Werkzeuge bereitstellen, um die Sachverhalte eines Anwendungsgebiets eindeutig abzubilden und zu beschreiben. Im Rahmen der Softwareentwicklung werden insbesondere Beschreibungssprachen eingesetzt, um Anforderungen an eine zu realisierende Software zu präzisieren oder um Entwurfs- und Softwarearchitekturbeschreibungen zu erstellen. Neben dedizierten Sprachen für spezifische Bereiche sind die *Unified Modeling Language*<sup>31</sup> (UML) und die *Extensible Markup Language*<sup>32</sup> (XML) die am weitesten verbreiteten und darüber hinaus standardisierten Beschreibungssprachen. XML ist eine ausschließlich textbasierte Sprache, während UML eine grafische Beschreibungssprache darstellt, die vorwiegend in der Softwaremodellierung eingesetzt wird. Infolgedessen wird UML auch als

<sup>31</sup> *Unified Modeling Language*, engl. Vereinheitlichte Modellierungssprache

<sup>32</sup> *Extensible Markup Language*, engl. Erweiterbare Auszeichnungssprache

Modellierungssprache bezeichnet, die in Bereichen der objektorientierten Analyse und des Entwurfs von Anwendung eingesetzt wird und dazu dient, Modelle eines Softwaresystems zu erstellen. Die UML bietet hierzu eine Reihe verschiedener Diagrammformen, die durch eine festgelegte Syntax beschrieben sind. XML ist, im Gegensatz zur UML, eine textuelle Beschreibungssprache. XML ist eine spezielle Ausprägung der Beschreibungssprachen, die als *deklarative Auszeichnungssprache* bezeichnet wird. Diese fungiert als Sprache, mit der neue Sprachen zur Darstellung hierarchisch strukturierter Daten beschrieben werden können. Die inhaltlichen Bestandteile eines XML-Dokuments werden auf Basis von *Markups* strukturiert und sind infolgedessen exakt spezifizierbar. Für die Trennung von Struktur und Inhalt wird in XML die *Document Type Definition (DTD)* oder *XML-Schema* eingesetzt. Für die Beschreibung beziehungsweise Modellierung von Softwaresystemen können die Sprachen XML und UML, entsprechend der zu vollziehenden Aufgabe und des Anwendungskontextes, auch gemeinsam im Entwicklungsprozess eingesetzt werden. Hierbei dienen *Metamodelle* der formalen Beschreibung der Vorgehensweise im Softwareentwicklungsprozess unter Verwendung von Beschreibungssprachen. Dabei ist zu berücksichtigen, welche Notationsform für eine Modellform geeignet ist, um den jeweiligen Verfahrensschritt konstruktiv und die Gesamtentwicklung effektiv und effizient zu vollziehen (vgl. [PUERTA 1997, S. 40–41]).

Mit modellgestützten Verfahren werden Vorgehensweisen beschrieben, die sich aus Teilschritten des Entwurfs und der Implementierung eines Interface zusammensetzen und ausgehend von einer abstrakten Beschreibung die Entwicklung einer konkreten Benutzerschnittstelle zum Ziel haben. Der Entwurf und die maschinenlesbare Beschreibung werden bei diesem Vorgehen auf Basis eines kombinierten Einsatzes von Modellen vollzogen, indem auf diesen eine Benutzerschnittstelle schrittweise spezifiziert, konkretisiert und anschließend durch Transformationen in eine Beschreibung – als Quelltext – auf eine Zielplattform überführt wird [KLEPPE U. A. 2003, KAP. 1]. Während in den abstrakten und konkreten Entwurfsphasen die UML eingesetzt wird, basiert der Quelltext einer 3D-Benutzerschnittstelle entweder auf Programmiersprachen oder auf Beschreibungssprachen, die überwiegend auf der Auszeichnungssprache XML aufbauen (vgl. [SOUSA U. A. 2007]). Die bestehenden Verfahren zur Unterstützung des Entwicklungsprozesses von Softwaresystemen werden begrifflich in *modellbasiert* und *modellgetrieben* unterschieden und sind im Kontext des Benutzerschnittstellenentwurfes nicht synonym zu verwenden, da diese sich vor allem darin unterscheiden, inwieweit die aufeinander aufbauenden Modelle (halb)automatisch ineinander überführt werden können (vgl. [WOLFF 2011, S. 6]). Bei der *modellgetriebenen Entwicklung* [STAHL U. A. 2007, KAP. 2] werden die Anwendungen, basierend auf Modellen, durch ein Programm generiert. Dazu werden die Modelle entweder unter Einbindung von Zwischenschritten zu konkreteren Modellen (Modell-zu-Modell-Transformation) oder in einem finalen Schritt in ein Programmgerüst (Modell-zu-Code-Transformation) transformiert. *Model Driven Architecture (MDA)* bezeichnet einen modellgetriebenen Entwicklungsansatz der *Object Management Group (OMG)* mit dem Fokus auf softwaretechnologische Entwicklungen. Der von der OMG vorgestellte Ansatz verfolgt die Vision von der Interoperabilität und Vielfalt von Softwaresystemen. Dabei liegt die Forderung nach einer Trennung von fachlicher Essenz (die Anwendersicht) und technischer Details (der Entwicklersicht) zugrunde. Die formalen Modelle sorgen hierbei für die notwendige Präzision. Dazu stehen innerhalb der Modellierungsschritte

verschiedene Diagrammtypen zur Verfügung, die auf Basis der UML-Notation einerseits der Beschreibung des Verhaltens, andererseits der Struktur des Systems dienen (vgl. [OMG]). Bei der modellgetriebenen Entwicklung werden aus plattformunabhängigen Modellen (*Computation/Platform Independent Model (CIM/PIM)*) durch Transformationen plattformspezifische Artefakte (*Platform Specific Model, PSM*) gebildet. Der Quellcode beziehungsweise das lauffähige interaktive System (*Platform Specific Implementation, PSI*) lässt sich im finalen Schritt teilweise oder komplett generieren (vgl. [PETRASCH & MEIMBERG 2006]). Der MDA-Ansatz umfasst daher nicht nur konzeptionelle, sondern ebenso maschinenlesbare Modelle, die durch Werkzeuge der rechnergestützten Softwareentwicklung gelesen werden können und eine weitgehend automatische Umsetzung auf verschiedenen Plattformen vollziehen.

Die *modellbasierte Entwicklung* beruht auf der Verwendung expliziter Modelle in den unterschiedlichen Phasen des Entwurfsprozesses für eine methodische Beschreibung einer zu realisierenden Benutzerschnittstelle. Im Rahmen dieses Ansatzes werden verschiedenen Aspekte eines grafischen Interface mit unterschiedlichen (meist) deklarativen Modellen abgebildet [SCHLUNGBAUM & ELWERT 1996, S. 7]. LIMBOURG und VANDERDONCK bezeichnen in ([LIMBOURG & VANDERDONCK 2004, S. 326]) die enge Verknüpfung des konzeptionellen und des softwaretechnologischen Interfacedesigns als „*multi path UI development*“<sup>33</sup> und beschreiben den modellbasierten Ansatz als Brücke zwischen den Disziplinen, die an der Entwicklung interaktiver Anwendungen beteiligt sind. Im Unterschied zur modellgetriebenen Entwicklung werden in modellbasierten Ansätzen die Modelle nicht zwingend in einem automatisierten Prozess ineinander überführt beziehungsweise im finalen Schritt in ein Programmgerüst transformiert beziehungsweise eine Anwendung generiert. Die Einhaltung der Vorgaben des Metamodells sowie der Einzelmodelle obliegt dem Entwickler und wird durch eingesetzte Softwarewerkzeuge unterstützt, da eine spätere Änderung eines Modells eine manuelle Anpassung der Einzelschritte in der Interfaceentwicklung zur Folge hätte. In den letzten Jahren hat sich vor allem das *Cameleon Referenzmodell* (vgl. [CALVARY U. A. 2003, S. 294 ff.]) als Metamodell für modellbasierte Vorgehensweisen etabliert. Das Referenzmodell für den Entwurf grafischer Benutzerschnittstellen beschreibt modellbasierte Verfahrensschritte, die von einer abstrakten zu einer konkreten Interfacebeschreibung führen. Die vier Abstraktionsebenen des Cameleon Referenzmodells sind *Aufgaben und Konzepte*, *Abstraktes Interface*, *Konkretes Interface* und *Finales Interface*. Dieses Modell wird beispielsweise für die Modellierung von Benutzerschnittstellen im Ansatz von LIMBOURG und VANDERDONCK mithilfe der XML-basierten, domänenspezifischen Sprache *User Interface eXtensible Markup Language (UsiXML)* angewendet [VANDERDONCK U. A. 2004, S. 329 ff.]. UsiXML ist eine plattformunabhängige Interfacebeschreibungssprache basierend auf den Ebenen des Cameleon Referenzmodells. Jedes in UsiXML beschriebene Modell kann separiert und der Entwurf einer Benutzerschnittstelle auf den unterschiedlichen Abstraktionsebenen expliziert werden.

Neben dem vorgestellten modellbasierten Ansatz existieren weitere Vorgehensweisen. Diese sind einerseits für konventionelle grafische 2D-Benutzeroberflächen (*UMLi*, vgl. [PINHEIRO DA SILVA & PATON 2003]), für ubiquitäre Systeme (*MARIA*, vgl. [PATERNO U. A. 2009]) und für 3D-Interfaces [WINGRAVE 2008; VITZTHUM 2008] ausgelegt. Es erfolgen an dieser Stelle keine detaillierten Ausführungen, weshalb auf die angegebenen Quellen verwiesen wird.

<sup>33</sup> *multi path UI development* engl. Benutzerschnittstellenentwicklung auf verschiedenen Wegen

### 2.3.2 Entwurfsmuster im Interfacedesign

*Entwurfsmuster* sind Lösungsschablonen für wiederkehrende Entwurfsprobleme sowohl in der Architektur als auch in der Softwareentwicklung und dem Interfacedesign. Den Ursprung haben Entwurfsmuster in der Mustersprache von CHRISTOPHER ALEXANDER, der diese folgendermaßen beschreibt:

*“Each pattern describes a problem which occurs over and over again in our environment, and then describes the core of the solution to that problem in such a way that you can use this solution a million times over, without ever doing it the same way twice.”*<sup>34</sup> [ALEXANDER U. A. 1977, S. X]

Das von ALEXANDER entwickelte Prinzip der Entwurfsmuster in einer Mustersprache wurde in vielen Bereichen außerhalb der Architektur aufgenommen und weiterentwickelt. Dabei haben Entwurfsmuster in der Softwareentwicklung eine weit größere Verbreitung erlangt als in der Architektur. Wurde zu Beginn der siebziger Jahre der Gedanke der Wiederverwendbarkeit von Quellcode praktisch realisiert, ist seit den 1990er Jahren, durch den Einsatz von Entwurfsmustern, die Software flexibel und wiederverwendbar implementiert und die Softwarekonstruktion mit Theorien und Konzepten untermauert (vgl. [GAMMA U. A. 1994]). Die Softwareentwurfsmuster unterscheiden sich strukturell von denen aus der Architektur. Die Entwurfsmuster der Softwaretechnologie basieren auf einer aufgabenbezogenen, nicht-hierarchischen Systematisierung und sind als Verbund von Mustern nicht in einer Mustersprache verknüpft. In der Mensch-Computer-Interaktion werden, speziell für grafische Benutzerschnittstellen, ebenfalls Entwurfsmuster eingesetzt. Die Entwurfsmuster im Interfacedesign sollen – vergleichbar mit Architekturmustern – bewährte Lösungen für wiederkehrende Problemstellungen bieten, die sich bei der Entwicklung interaktiver Systeme ergeben (vgl. [BORCHERS 2000A, S. 1]). Der Fokus von Entwurfsmustern leitet sich dabei weniger von softwaretechnologischen sondern von softwareergonomischen Anforderungen ab, wie diese beispielsweise in der ISO9241 (vgl. [DIN 2002]) festgehalten sind. Das Spektrum an Entwurfsmustern, derer es im Bereich der Benutzerschnittstellenentwicklung inzwischen zahlreiche Sammlungen gibt, reicht von ausschließlich textueller Beschreibung bis zum Einsatz formaler Sprachen. Daher werden nunmehr wesentliche Entwurfsmustersammlungen und Mustersprachen des Interfacedesigns benannt. In [TIDWELL 2011] wird eine lose Sammlung von Entwurfsmustern vorgestellt, die durch fehlende Verbindungen und Abhängigkeiten zwischen den Entwurfsmustern vornehmlich erfahrene Entwickler adressiert. Diese lose Zusammenstellung umfasst eine sehr große Anzahl an Entwurfsmustern und bietet Vorlagen für verschiedene Bereiche der Mensch-Computer-Interaktion. Ein nutzerorientierter Ansatz wird in [FOLMER U. A. 2003, S. 81 ff.] gewählt, in dem Erkenntnisse zur Benutzerfreundlichkeit aus verschiedenen Quellen zusammengefasst und in Form abstrakter Entwurfsmuster zu einer Mustersprache aufgearbeitet werden. Die Autoren erläutern durch die Darstellung von Zusammenhängen die Wechselwirkung zwischen Softwaresystem und Softwareergonomie ohne Aspekte der Implementierung oder der prozessinternen Kommunikation zu betrachten. Eine weitere Möglichkeit ist eine Strukturierung von Entwurfsmustern nach dem Vorbild der

<sup>34</sup> „Jedes Entwurfsmuster beschreibt ein Problem das in unserer Umgebung wiederkehrend auftritt und beschreibt den Kern der Lösung für dieses Problem auf eine Art und Weise, dass diese Lösung millionenfach angewendet werden kann, ohne dass dies jemals in derselben Form erfolgt.“

Architekturmuster. Eine mit der Mustersprache von ALEXANDER vergleichbare hierarchische Strukturierung wird in [VAN WELIE & VAN DER VEER 2003, S. 528 ff.] zur Entwicklung von Benutzerschnittstellen vorgestellt. Dabei wird eine Skalierung von Herausforderungen im Interaktionsdesign zugrunde gelegt. Zunächst erfolgt durch eine abstrakte Beschreibung eine grundsätzliche Konzeption, bevor in konkreten Ausführungen die Ausgestaltung der Benutzerschnittstelle festgelegt wird. Einen vergleichbaren Ansatz verfolgt DUYNE U. A. in [DUYNE U. A. 2003, KAP. 2] mit einer Mustersprache für den Entwurf von Webseiten. Die Anwendung der beschriebenen Entwurfsmuster ist an die Vorgehensweise im Entwurfsprozess von Benutzerschnittstellen angelehnt und auf die in diesem Verlauf auftretenden Probleme ausgerichtet.

## 2.4 Spannungsfeld der Technischen Visualistik

Die aktuellen Entwicklungen auf dem Gebiet der interaktiven 3D-Anwendungen zeigen, dass die von VILÉM FLUSSER im Jahr 1985 formulierte Zukunftsvision der „virtuellen Realität“ sich bereits in Teilen erfüllt hat (vgl. [FLUSSER 1999]). Eine virtuelle 3D-Szene wird mithilfe einer virtuellen Kamera in ein planares Abbild überführt. Die dabei eingesetzten Abbildungsverfahren scheinen in der Tradition, der seit dem Beginn der Renaissance fortschreitenden perspektivischen Verbildlichung, zu stehen. Jedoch verharret die Computergrafik, durch die Nachahmung der *Camera obscura*, zu oft im Streben nach einer Reproduktion der realen Welt. Dabei ist der Computer nicht an die Grenzen des Fotoapparates beziehungsweise der Filmkamera gebunden. Im Unterschied zum Abbildungsvorgang einer dreidimensionalen Umgebung mit einer realen Kamera ist der Ausgangspunkt computergrafischer Bilder das zugrunde liegende Datum, das durch eine flexibel formbare Bildsynthese als technisches Bild verwirklicht wird. Durch einen mathematisch-maschinellen Vorgang und ohne ein beeinflussendes oder sogar kreatives Zutun ist dieser Prozess jedoch eine berechnungsoptimierte Abarbeitung von Algorithmen. Im Hinblick auf die konventionellen computergrafischen 3D-Anwendungen erfolgt diese Verbildlichung durch die etablierten Verfahren zur Darstellung des Raumes; die Parallelprojektion und die Zentralprojektion. Im Gegensatz dazu bietet eine freie Parameterwahl im Prozess der Bildsynthese eine potenziell-unbegrenzte Vielfalt an computergrafischen Bildern (vgl. [SALOMON 2006, S. 145]). Durch die etablierten Abbildungstechniken computergrafischer Anwendungen einerseits und einem außerordentlichen Repertoire an Darstellungsmöglichkeiten andererseits vergegenständlicht sich eine Diskrepanz bei der Ausgestaltung der visuellen Kommunikation in computergrafischen Anwendungen. Diese Auseinandersetzungen mit technischen Bildern beschreibt FLUSSER als *„Resultate eines verzwickten Kampfes zwischen den Erfindern und den Kontrolleuren der Apparate, einer Zusammenarbeit zwischen beiden und eines Kampfes und einer Zusammenarbeit zwischen Apparaten und Menschen“* [FLUSSER 1999, S. 26]. Von Ingenieuren („Erfindern“) gebaute (Kamera-)Modelle erzeugen Abbilder in einem automatischen Vorgang. In diesen Bilderzeugungsprozess greift der Designer („Kontrolleur“) ein (vgl. [FLUSSER 1999, S. 24]). Dies resultiert im Bereich interaktiver 3D-Anwendungen in einem Spannungsfeld zwischen den Erfindern und den Kontrolleuren einerseits und zwischen der konventionellen und der schöpferischen Bilderzeugung andererseits.

In diesem Spannungsfeld sind die Tätigkeitsbereiche von Interfacedesignern, Computergrafikern und Softwareentwicklern verortet. MITCHELL KAPOR beschreibt in seinem Artikel „*A Software Design Manifesto*“ [KAPOR 1996, KAP. 1] die Verwirklichung von Softwareprodukten als eine Analogie zum Bau eines Gebäudes. Unter der Berücksichtigung von Anforderungen liegt die Bauplanung und –durchführung in der Verantwortlichkeit des Architekten, während der Bauingenieur die Vorgaben unter Berücksichtigung der technischen Umsetzbarkeit vollzieht. Architekten entwerfen folglich die Gebäude, mit denen die Bewohner direkt „interagieren“, indem sie darin leben. Benutzerschnittstellen interaktiver Anwendungen werden ebenfalls entworfen, damit der Nutzer mit dem System interagieren kann. KAPOR zieht daraus den Schluss, dass der Interfacedesigner in der Softwareentwicklung eine dem Architekten vergleichbare Stellung einnimmt. Die Rolle des Softwareingenieurs, der die Implementierung und die funktionale Struktur einer interaktiven Anwendung sicherstellt, ist hingegen mit der Rolle eines Bauingenieurs vergleichbar.

Gegen KAPORS Argumentation ist möglicherweise einzuwenden, dass der Gebäudebau und die Anwendungsentwicklung in der Prozessualität zu unterschiedlich seien, als dass die vorgeführte Analogie nachvollziehbar sei. Ungeachtet dessen ist die von KAPOR dargelegte begriffliche Trennung von Design und Entwicklung, als zwei ineinandergreifende Aspekte der Entwicklung interaktiver 3D-Anwendungen, plausibel. Im Rahmen eines nutzerfokussierten Gestaltungs- und Entwurfsprozesses stehen insbesondere der Entwurf und die Etablierung eines interaktiven visuellen Dialoges zwischen dem Anwender und dem System im Vordergrund. Dabei müssen jedoch bestehende technische und technologische Rahmenbedingungen nachvollzogen und berücksichtigt werden. Im Bereich der Softwaretechnik stehen hingegen die Konzeption, der Entwurf und die Implementierung des Systems im Mittelpunkt, um eine fehlerfreie Funktionalität einer Anwendung sicherzustellen. Gleichzeitig ist eine möglichst unverfälschte Umsetzung der Designkonzepte notwendig.

Für die Unterstützung von Ingenieuren (die Erfinder) und Gestaltern (die Kontrolleure) ist es einerseits zweckmäßig den Entwurfsprozess von 3D-Benutzerschnittstellen in eine methodische Vorgehensweise zu überführen, andererseits die softwaretechnologischen Werkzeuge zur Integration einer systematisierten visuellen Vielfalt computergrafischer Darstellungen in 3D-Anwendungen bereitzustellen.



### 3 Abbildungsverfahren für interaktive 3D-Anwendungen

Die Entwicklungen in der 3D-Computergrafik werden seit jeher durch das Streben nach dem computergrafischen Fotorealismus beeinflusst. Nach über 40 Jahren der Wissenschaft und Forschung ist die Mehrzahl der Herausforderungen gelöst, die bei der computergrafischen Reproduktion eines fotografischen Abbildes bestanden. Abseits der Ähnlichkeitsparadigmen und des Naturalismus entstehen jedoch Bilder, die nicht auf einer konventionellen Bildsynthese beruhen. In der 3D-Computergrafik finden diese methodischen Abweichungen von der präzisen Nachbildung vorwiegend im Forschungsbereich nicht-fotorealistischer Darstellungsverfahren eine Beachtung. GOOCH und GOOCH verdeutlichen in [GOOCH & GOOCH 2001, S. 1-3], ebenso wie STROTHOTTE u. A. in [STROTHOTTE & SCHLECHTWEG 2002, S. 7 ff.], dass computergrafische Darstellungsweisen es durch den Verzicht auf eine genaue Nachbildung des fotooptischen Abbildungsprozesses vermögen, komplexe Sachverhalte dem Betrachter in 3D-Anwendungen besser zu vermitteln. Im Sinne einer „*Abbildbarkeit situativer und interaktiver Relationen der Nutzer [...] zu ihren Interaktionsobjekten*“ [GROH 2011, S. 8] ist es daher zweckdienlich, wenn die 3D-Computergrafik und die Bildgestaltung in die gleiche Richtung streben. Die bestehenden Verfahren aus dem Fachbereich des Non-Photorealistic Rendering zeigen in deren Funktion als Mittel der Gestaltung einerseits, dass ein vielseitiges Spektrum existiert, andererseits eine zielgerichtete Bereitstellung von adaptierbaren Darstellungstechniken eine Verwirklichung von effizienten grafischen Benutzerschnittstellen sicherstellt. Mit einem Blick auf bestehende 3D-Anwendungen stellt sich jedoch die Frage, warum diese Vielfalt als Gestaltungsmittel bisher weitgehend unbeachtet geblieben ist. DURAND sieht in der desorganisierten Mannigfaltigkeit der Verfahren eine wesentliche Ursache, da infolgedessen Defizite in der Zuordnung von Technik und Darstellungsintention sowie in der anwendungsbezogenen Systematisierung der Methoden bestehen [DURAND 2002A, S. 111]. Ferner fordert DEUSSEN diesbezüglich „*Regeln für eine ‚gute‘ Darstellung (...) [bei denen] dynamische situationsbezogene Darstellungsarten eine wichtige Rolle spielen*“ [DEUSSEN 2001, S. 79]

Zunächst werden in diesem Kapitel ein historischer Abriss computergrafischer Interfaces gegeben und Einsatzmöglichkeiten sowie Beispiele vorgestellt (3.1). Nachfolgend einer Betrachtung von Darstellungsweisen abseits der Computergrafik (3.2) wird für eine Aufarbeitung von 3D-Darstellungsmöglichkeiten das computergrafische Kameramodell erläutert und dessen Modifikationsmöglichkeiten zur Gestaltung von Bildräumen analysiert (3.3). Basierend auf den Erkenntnissen und daraus hervorgehenden Gestaltungshinweisen (3.4) werden die bildstrukturellen Eigenschaften computergrafischer Abbilder kategorisiert und im Hinblick auf die Visualisierungsziele, unter Einbeziehung existierender Verfahren und deren Gebrauch im Kontext der Gestaltung von 3D-Benutzerschnittstellen, systematisiert (3.5).

## 3.1 Computergrafische Benutzerschnittstellen

### 3.1.1 Historischer Abriss

#### Entwicklungen in den 1950er und 1960er Jahren

In den 1950er Jahren werden Computer erstmalig mit Monitoren ausgestattet, wodurch die Grundlage für die ersten computergrafisch erzeugten Abbilder bereitet wird. Zum Ende des Jahrzehnts, im Jahr 1959, wird mit dem *DAC-1* das erste Computer-Zeichnungssystem entwickelt (vgl. [MASSON 2007, S. 386]) und im Jahr 1964 der Öffentlichkeit präsentiert. Im selben Jahrzehnt entwickelt IVAN SUTHERLAND mit *Sketch pad* ein System zur Verwaltung von Daten mithilfe von Kompositionen aus grafischen Standardelementen (vgl. [SUTHERLAND 1964]). Die Charakteristiken des Interface stellen sich als ein entscheidender Schritt in der Entwicklung der interaktiven computergrafischen Benutzerschnittstellen heraus. Ferner stellen EVANS und SUTHERLAND im Jahr 1969 das erste kommerzielle CAD-Drahtgitter-Zeichnungssystem, *LDS-1*, vor. Dieses basiert auf dem Verfahren der Drahtgitterdarstellung aus SUTHERLANDS früherer Arbeit. Weiterhin entwickelt ALAN KAY im selben Jahr die Metapher einer grafischen Benutzerschnittstelle, die als Vorbild für das Design der Benutzeroberfläche des Apple Macintosh dient. Ebenfalls im Jahr 1969 wurde die *Special Interest Group on Graphics (SIGGRAPH)* [ACM] der *Association for Computing Machinery (ACM)* gegründet. Bis heute ist die Organisation eine der einflussreichsten Vereinigungen im Bereich Forschung und Entwicklung von computergrafischen Darstellungen und interaktiven Benutzerschnittstellen.

#### Entwicklungen in den 1970er Jahren

Die Einführung des *Personal Computers* in den 1970er Jahren hat weitreichende Auswirkungen auf die Entwicklung interaktiver Systeme (vgl. [MACHOVER 1994, S. 16]). Die zunehmende Verbreitung und Segmentierung des Personal Computers in unterschiedlichen Anwendungsbereichen führt zu einer herstellerseitigen Intensivierung der Hardwareentwicklung. Dadurch werden in den 1970er Jahren neue Möglichkeiten in der Mensch-Maschine-Kommunikation, beispielsweise *Trackball*<sup>35</sup> und *Grafiktablets* als Eingabegeräte sowie weiterentwickelte Rasterbildschirme als Ausgabegeräte, bereitstellt. Auch im Softwarebereich werden wesentliche Entwicklungen, vor allem im Bereich der Bildsynthese, vollzogen. Viele der heute eingesetzten Algorithmen zur Beleuchtung und Texturierung, beispielsweise *Gouraud*- und *Phong-Shading* sowie *Bump-Mapping*, stammen aus den 1970er Jahren. Die Algorithmen sind als Verfahren in die modernen Grafikkarten integriert und werden in den Abbilderzeugungsprozess eingebunden (vgl. [AKENINE-MÖLLER U. A. 2008, S. 110 FF. UND 183 FF.]). Diese Möglichkeiten in der Bildgestaltung führten vor allem unter dem Einsatz von fotorealistischen Texturen, Spiegelungen und Reflexionen zu Veränderungen im 3D-Interfacedesign und finden Eingang in die Bereiche der Kunst und der Unterhaltung. Beispielsweise entsteht 1976 mit *Futureworld* der erste Spielfilm, der das computergrafische Modell einer Hand und eines Kopfes zeigt (vgl. [MASSON 2007, S. 401]).

---

<sup>35</sup> Ein Eingabegerät, bei dem eine Kugel, die mit den Fingern bewegt wird, an einem starren Element angebracht ist, wird als Trackball bezeichnet.



## Entwicklungen in den 1980er Jahren

In den 1980er Jahren ereignen sich die Gründungen vieler Unternehmen, beispielsweise Adobe, Autodesk und Sun Microsystems, die bis heute einen bedeutsamen Bezug zu computergrafischen Darstellungen und interaktiven Anwendungen besitzen. Die entwickelten Computersysteme dieser Unternehmen arbeiten zunächst auf Großrechnern, die ab Mitte der 1980er Jahre allmählich durch Workstations ersetzt werden. Mit einer stetig steigenden Systemleistung von Personal Computern und Workstations erhalten zunehmend Endanwender einen Zugang zu interaktiven Grafikanwendungen. Bis zum Ende des Jahrzehnts sind interaktive Grafikapplikationen für eine große Anzahl von Anwendungsbereichen verfügbar. Durch ANSI und ISO wird im Jahr 1985 die erste 2D-Computergrafik-Norm verabschiedet. Diese legt für Anwendungsprogramme die Mindestanforderungen fest, um die Entwicklung von grafischen Softwaresystemen zu vereinfachen und gleichzeitig das effektive Zusammenwirken von Hard- und Software sicherzustellen. Darüber hinaus bildet sich der erste Farbgrafikstandard heraus (CGA), der in den Folgejahren durch EGA (1984) und VGA (1987) ersetzt wird. Ferner etablierten sich Industriestandards, beispielsweise die Seitenbeschreibungssprache *PostScript* von Adobe oder *X Window System*, für grafische Schnittstellen. Im Bereich der 3D-Computergrafik folgen ebenfalls Normen und Standards, die von Weiterentwicklungen im Bereich computergrafischer Visualisierungstechniken begleitet werden. In den 1980er Jahren sind vor allem das *Raytracing*- und das *Radiosity-Verfahren* hervorzuheben, die wesentliche Methoden zur Berechnung der Lichtverteilung in computergrafischen Szenen sind und eine naturalistische, globale Beleuchtung für virtuelle Umgebungen simulieren (vgl. [AKENINE-MÖLLER U. A. 2008, S. 409 ff.]). Die neuen Entwicklungen im Hard- und Softwarebereich haben zunehmend Einfluss auf die Bereiche Kunst und Unterhaltung. Aufgrund der Entwicklungen in den 1980er Jahren werden die ersten 3D-Videospiele, beispielsweise *Cube Quest* von Simutrek, realisierbar.

## Entwicklungen in den 1990er Jahren

In den 1990er Jahren bietet die verfügbare Hardware die Möglichkeit, Bereiche der Computergrafik mit verwandten Gebieten, beispielsweise die Bildverarbeitung, zusammenzuführen. Bis zu diesem Zeitpunkt basierten die Anwendungen der Fachbereiche auf unterschiedliche Workstation-Architekturen. Ist der überwiegende Teil der computergrafischen Anwendungen zu Beginn des Jahrzehnts weiterhin in wissenschaftlichen und technischen Bereichen eingesetzt, verlagert sich im Laufe der 1990er Jahre der Fokus zunehmend auf verschiedene Bereiche der Unterhaltung (vgl. [MACHOVER 1994, S. 18]). Dadurch etablierten sich Entwurfs- und Visualisierungswerkzeuge in immer mehr Anwendungsbereichen, wie etwa in der Filmindustrie durch Animations- und Rendersoftware oder in der Videospiegelbranche durch interaktive Level-Editoren. Im Jahr 1992 wird die 3D-Grafik-Programmierschnittstelle OpenGL vorgestellt und bietet in der Folge eine Schnittstelle für die Entwicklung und die Verbreitung plattform- und domänenunabhängiger computergrafischer Anwendungen. Bereits vier Jahre später erfolgt die Einführung des ersten Grafikprozessors für Endanwender, der eine 3D-Hardwarebeschleunigung bietet. In den späten 1990er Jahren erfolgt, aufgrund der zur Verfügung stehenden Rechenleistung und dem Bestreben nach naturalistischen Darstellungen, die Einbindung von physikalischen Gegebenheiten wie beispielsweise Wasser und Wind

in Simulationsanwendungen und Computerspiele. Weiterhin werden zunehmend realitätsnachahmende Animationen, beispielsweise in Simulations- und Präsentationsanwendungen sowie im Bereich der Computerspiele, eingesetzt, die dem Anwender vielfältige Interaktionen mit computergenerierten, interaktiven 3D-Umgebungen ermöglichen.

### Entwicklungen in den 2000er Jahren

Die Entwicklungen von interaktiven computergrafischen Anwendungen im ersten Jahrzehnt des 21. Jahrhunderts sind überwiegend durch den Fortschritt im Bereich des *Grafikprozessors* (*GPU*)<sup>36</sup> geprägt. Grafik-Workstations werden immer häufiger von Computern mit einer dedizierten Grafikkarte abgelöst. Die GPU wird zum schnellen Co-Prozessor des Computers, wodurch es möglich wird, bisherige Offline-Techniken und Verfahren für den Anwender in Echtzeitsystemen bereitzustellen. Profiteur dieser Entwicklung ist vor allem die Spieleindustrie, die in den 2000er Jahren eine starke grafische Weiterentwicklung erfährt. Mittels programmierbarer Schattierer, die Darstellungseffekte in der 3D-Computergrafik implementieren, können vielfältige naturalistische Darstellungen und Effekte in interaktive Anwendungen integriert werden. Dabei liegen die Schwerpunkte der Entwicklung auf Beleuchtungsmodellen, verbesserten Texturierungsmöglichkeiten und der Verfeinerung von Details in Geometrie und Gestalt räumlicher Szenen, wodurch der Grad an „Wirklichkeit“ im 3D-Benutzerschnittstellen zunimmt. Gleichzeitig bietet die zunehmende Rechenleistung die Möglichkeit interaktive 3D-Anwendungen zu entwickeln, mit denen ein Anwender eine Interaktion nicht auf Basis einer symbolischen Handlungen oder der Handlung mit Symbolen, sondern mit der virtuellen Umgebung und den Elementen des Objektraums selbst ausführt.

Die Entwicklungen computergrafischer Darstellungen in interaktiven 3D-Anwendungen sind bis heute überwiegend von den Hardwareentwicklungen beeinflusst. Die Veränderungen im Softwarebereich erfolgen in einem nachgelagerten Schritt durch die Einführung von Methoden und Verfahren, die durch die Entwicklungen im Hardwarebereich umsetzbar werden.

### 3.1.2 Anwendungsdomänen

Der historische Abriss im vorangegangenen Abschnitt verdeutlicht die stetig zunehmende Grafikleistung, woraufhin sich zahlreiche Anwendungsdomänen für interaktive 3D-Anwendungen herausbilden konnten. Vor allem die Darstellung realer Objekte in Bereichen der Architektur, Produktpräsentation und Unterhaltung aber auch die Verbildlichung abstrakter Daten – wenn diese a priori räumlich sind – stehen im Mittelpunkt der Anwendungsentwicklung. Dieser Abschnitt gibt einen Überblick über Einsatzbereiche von interaktiven 3D-Anwendungen. Der Querschnitt verfügbarer Softwaresysteme erstreckt sich über die Bereiche E-Commerce und Unterhaltung, Lehren, Lernen und Trainieren sowie Kultur und Tourismus. Die folgende Zusammenstellung gibt einen schlaglichtartigen Einblick in die vielfältigen Anwendungsfelder, wobei sich die Bereiche in Teilen überlappen. Dabei werden die Domänen vor allem unter dem Aspekt der Verwendung von interaktiven 3D-Anwendungen betrachtet und das Potenzial sowie die aktuelle Verbreitung dieser aufgezeigt.

---

<sup>36</sup> *Graphics Processing Unit (GPU)* engl. Grafikprozessor; dient zur Berechnung der Bildschirmausgabe auf Computern

### 3.1.2.1 E-Commerce und Unterhaltung

Für die visuelle Präsentation und Konfiguration von Produkten oder Produktgruppen werden zunehmend 3D-Anwendungen eingesetzt, um den Kunden die Möglichkeit zu geben, ein Produkt interaktiv zu erfahren. Die Anwendungen stellen für viele Unternehmen eine ideale Kommunikationslösung für das Marketing und die Werbung dar, weshalb die Systeme vor allem auf einen kommerziellen Nutzen ausgerichtet sind. Die 3D-Produktdarstellungen werden eingesetzt, wenn eine Ware komplexe Funktionen aufweist, erklärungsbedürftig ist oder über zahlreiche Varianten oder Konfigurationsmöglichkeiten verfügt. Zu den typischen Branchen, die interaktive 3D-Anwendungen für die Produktpräsentation einsetzen, gehören die Medizintechnik, die Bau-, die Möbel-, die Automobil- und die Maschinenindustrie. Die Produktpräsentationen erfolgen, indem vorwiegend proprietäre, interaktive Applikationen das Produktportfolio eines Unternehmens erfahrbar machen. Die interaktiven Konfiguratoren ermöglichen beispielsweise eine individuelle Gestaltung von Fahrzeugen ([@AUDI AG], Abbildung 3-B) oder Maschinen beziehungsweise Maschinenbestandteilen, um ein Produkt an die Anforderungen des Kunden anzupassen. Weiterhin dienen interaktive 3D-Einrichtungsplaner im Wohn-, Küchen- und Badbereich für den virtuellen Entwurf des realen Eigenheims. Die Darstellungen bedienen sich in dieser Anwendungsdomäne vorwiegend einer zentralprojektiven Abbildung, um einen möglichst räumlichen Eindruck eines Handelsgutes zu vermitteln. Der Einsatz parallelprojektiver Visualisierungen wird darüber hinaus eingesetzt, um Charakteristiken und Eigenschaften in Form und Gestalt von Produkten hervorzuheben und als Kaufargument zu betonen.

Neben Produktpräsentationen ist der Unterhaltungssektor, der im Hinblick auf Visualisierungen vor allem durch Computerspiele dominiert wird, einer der wesentlichen Einsatzbereiche von 3D-Anwendungen. Aktuelle Computerspiele bieten frei-begehbare, virtuelle Welten, in denen die Nutzer spielerisch gegeneinander antreten oder durch eine Handlung geführt werden und diese interaktiv erleben. Da die Handlungsverläufe oftmals cineastisch inszeniert sind, wird in den Computerspielen eine dem Film nachempfundene Darstellungsweise von den Spielern eingefordert. Die 3D-Computerspiele, allen voran *First-Person Shooter*, beispielsweise die *Battlefield*-Reihe (Abbildung 3-A) des Entwicklerstudios Dice [@DICE] und *Third-Person Action Adventures*, zum Beispiel *Grand Theft Auto* der Firma Rockstar [@ROCKSTAR GAMES], erfüllen die Anforderung einer fotorealistischen Darstellungsweise von virtuellen 3D-Welten, um dadurch den wirtschaftlichen Erfolg sicherzustellen.

### 3.1.2.2 Lehren, Lernen und Trainieren

Die Tätigkeiten des Lehrens, Lernens und Trainierens werden zunehmend unter Einsatz interaktiver 3D-Anwendungen vollzogen, da sich mithilfe interaktiver 3D-Anwendungen Sachverhalte anschaulich beschreiben lassen und Abläufe zu Übungs- und Erkenntniszwecken nachgebildet werden können. Es existieren zahlreiche Applikationen für Bau- und Reparaturanleitungen beziehungsweise virtuelle Trainingsszenarien ([@NEOAxis], Abbildung 3-C), mit denen Fachkräfte verschiedener Berufszweige geschult werden und eigenständig trainieren können. Darüber hinaus werden bei der Simulation von realen Vorgängen

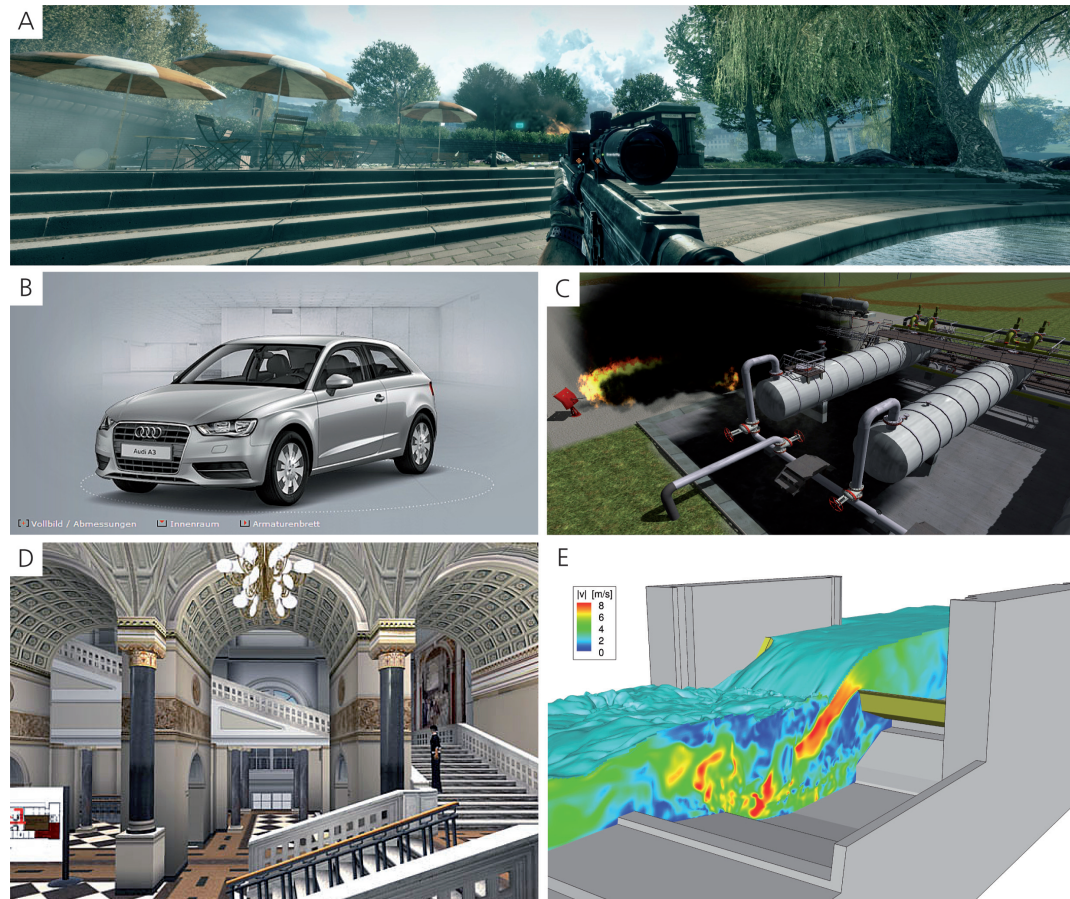


Abbildung 3: Benutzerschnittstellen interaktiver 3D-Anwendungen aus den Anwendungsbereichen Unterhaltung (A), E-Commerce (B), Lehr-/Lern- und Trainingsanwendungen (C), Kultur und Tourismus (D) und Datenvisualisierung (E)

computergrafische Darstellungen eingesetzt, um das Verhalten der Nutzer bei berufsbedingten Handlungsabläufen, beispielsweise in Flugsimulatoren, oder bei Verfahrensschritten in Notfallsituationen, zum Beispiel bei der Personenrettung, zu schulen. Die 3D-Anwendungen im Bereich des Lehrens, Lernens und Trainierens sind jedoch nicht auf ingenieurtechnische und trainingsrelevante Aspekte beschränkt. Die Systeme werden in Bereichen der Naturwissenschaften wie beispielsweise in der Chemie für die Veranschaulichung der Zusammenhänge von Stoffen oder in der Physik zur Darstellung thermischer oder elektrischer Vorgänge eingesetzt. Hierbei dienen die 3D-Anwendungen der Visualisierung abstrakter Daten, wobei die Veranschaulichung von Simulationen ([@BAW], Abbildung 3-E), die grafische Auswertung von Informationen und die Präsentation von Analysen konkrete Anwendungsszenarien darstellen. Interaktive 3D-Anwendungen erlauben die visuelle Exploration von Datenmengen und die effiziente Navigation in diesen. Ebenso können statistische Auswertungen zu politischen Geschehnissen und wirtschaftlichen Verflechtungen in interaktiven Anwendungen vollzogen werden. Dies geschieht, indem der Anwender durch die grafische Darstellung von Relationen und Informationen in seinem Verständnis für den Anwendungskontext unterstützt wird. Für die Visualisierung der Datencharakteristiken werden im Bereich der Informationsvisualisierung und der wissenschaftlichen Visualisierung vergleichsweise häufig Darstellungsweisen abseits der Zentralprojektion eingesetzt. Es werden nichtlineare Verfahren in den

Bereichen der Systembiologie und der Medizin aber auch im Ingenieurwesen ebenso wie den Geowissenschaften eingesetzt, um die durch eine zentralprojektive Darstellung nicht sichtbaren beziehungsweise verdeckten Informationen indes durch nicht-fotorealistische Abbildungsverfahren zu visualisieren. Dies ist möglich, da die Gestalt und die Form der verbildlichten Daten in dieser Anwendungsdomäne nicht zwingend die wesentlichen Charakteristiken sind, die mit einer Darstellung herausgestellt werden sollen.

### 3.1.2.3 Kultur und Tourismus

Die Anwendungsdomäne Städte- und Landschaftsvisualisierung ist prädestiniert für räumliche Darstellungen in interaktiven Anwendungen. Die realistische Visualisierung von digitalen Geländemodellen beziehungsweise von Gebäuden und Infrastrukturen sind wichtige Hilfsmittel und Vermarktungswerkzeuge für die Städteplanung, für Navigations- und Lokalisierungsdienstleister sowie für die Tourismusbranche. Besonders wichtig sind dabei die detailgetreuen Visualisierungen der oftmals komplexen 3D-Modelle. Diese werden entweder in einer einzelnen 3D-Ansicht aber oftmals auch in mehreren gleichzeitigen Sichten dargeboten, um die perspektivische Darstellung einer Umgebung mit Draufsichten und Panoramasichten zu kombinieren und zu ergänzen. Interaktive Anwendungen wie *Google Earth* [G00] oder Apples Dienst *Karten* ([A]) bieten 3D-Darstellungen, die dem Anwender die gesamte Erde als frei erkundbare Welt zur Verfügung stellen. Die Visualisierungen erreichen durch hochauflösende Texturen und komplexe Beleuchtung eine fotorealistische Darstellung und vermitteln infolgedessen einen realistischen Eindruck der tatsächlichen Umgebung. Dem Anwender wird dadurch, unabhängig vom jeweiligen Standort, die Möglichkeit gegeben, in einer fremden Umgebung ein Ladengeschäft zu finden, das Umfeld eines Hotels am Urlaubsort oder beispielsweise die nächstgelegene Bushaltestelle am Zielort zu identifizieren. Für Navigationssysteme werden ebenfalls 3D-Städte und Landschaftsvisualisierungen, unter zusätzlicher Einbindung von Informationen zu Sehenswürdigkeiten und Einkaufsmöglichkeiten, eingesetzt (vgl. [G]). Ein überwiegender Teil dieser Anwendungen bietet eine perspektivische Darstellung aus der Fahrersicht oder aus einer Verfolgerperspektive, um den Nutzer bei der Wegführung zu unterstützen. Darüber hinaus bieten einige der 3D-Anwendungen zusätzliche Darstellung, die über eine reine Nachbildung der Umgebung hinausgeht, um zusätzliche Informationen in schwierigen Verkehrssituationen, beispielsweise Kreuzungsbereiche oder Autobahnabfahrten, bereitzustellen. Weiterhin ist in dieser Anwendungsdomäne eine stetige Weiterentwicklung von interaktiven 3D-Anwendungen mit kulturellem oder sozialem Hintergrund zu verzeichnen. So bieten Applikationen, in denen Rundgänge durch virtuelle Museen möglich sind, eine Nachbildung bestehender Ausstellungsräume unter Berücksichtigung einer korrekt nachgebildeten Beleuchtung und Texturierung. Als eine der größten und bekanntesten Gemäldegalerien ist es die staatliche Kunstsammlung Dresden, die mit der *Galerie Alte Meister* eine der renommiertesten Gemäldesammlungen der Welt in der virtuellen Umgebung *Second Life* zugänglich macht (vgl. [L]) Abbildung 3-D).



### 3.1.3 Beispiele für interaktive 3D-Benutzerschnittstellen

Nachfolgend wird anhand von interaktiven Anwendungen das Potenzial von 3D-Benutzerschnittstellen aufgezeigt. Die vorgestellten 3D-Systeme werden im medizinischen Bereich, zur Darstellung abstrakter Daten sowie in Architektur- und Landschaftsvisualisierungen eingesetzt. Die an dieser Stelle getroffene Auswahl der Software erfolgt unter der Prämisse, beispielhaft 3D-Anwendungen für den Desktopbereich vorzustellen. Weiterhin sind deren Interfaces durch einen räumlichen Charakter geprägt und bedienen sich unterschiedlicher Abbildungstechniken zur Verbildlichung von Daten.

#### 3.1.3.1 Benutzerschnittstellen für wissenschaftliche und medizinische Veranschaulichungen

In vielen Bereichen der Wissenschaft werden Illustrationen genutzt, um Inhalte und Wissen effizient zu vermitteln. Beispielsweise werden in der Medizin illustrative Darstellungen eingesetzt, um den Aufbau und die Funktionsweise von Lebewesen und deren Bestandteile zu veranschaulichen. Auch in verwandten Anwendungsgebieten, beispielsweise der Biologie, dienen Illustrationen der Erläuterung von Sachverhalten. Diese Form der Darstellung findet in den letzten Jahren zunehmend Einsatz in interaktiven 3D-Anwendungen, da illustrative NPR-Techniken in den Bildsynthesevorgang integriert und als Visualisierungsform in 3D-Anwendungen bereitgestellt werden.

Die Software *INTERACTORIUM* der Universität von New South Wales ist eine interaktive Anwendung für wissenschaftliche Visualisierungen im Bereich der Systembiologie. Das Softwarewerkzeug basiert auf dem Projekt *Skyrails* [SKYRAILS] und dient der Darstellung von Proteinen und deren Reaktionen auf biochemische Abläufe und Vorgänge (vgl. [WIDJAJA U. A. 2009]). Die Anwendung bietet die Möglichkeit komplexe Netzwerke mit bis zu 40.000 Proteinen oder 6.000 Multiproteinkomplexen zu veranschaulichen (siehe Abbildung 4, links). Die Proteine können hierbei individuell extrahiert und in separaten Detailansichten exploriert werden. Die Anwendung INTERACTORIUM verfügt über verschiedene Interaktionsmöglichkeiten, die eine Manipulation der Daten und einen Eingriff in deren Darstellungsweise in Echtzeit bieten. Dabei ist es dem Anwender freigestellt, ohne eine systemseitige Unterstützung oder mittels eines geführten Interaktionsprozesses mit den visuellen Strukturen zu interagieren.

*MeVisLab* ist eine 3D-Anwendung im medizinischen Bereich, die von der MeVis Research GmbH als Forschungs- und Entwicklungsumgebung entwickelt wird. Das Softwaresystem ermöglicht die schnelle Realisierung prototypischer 3D-Anwendungen, die auf spezifische medizinische Anwendungsfälle adaptiert werden können (vgl. [MEVISLAB]). Dadurch können im klinischen Umfeld unterschiedliche Arbeitsabläufe und Behandlungsmethoden erprobt werden. Neben Softwaremodulen für medizinische Bildverarbeitungstechniken zur Segmentierung und quantitativen und funktionellen Analyse und Auswertung von Daten sind Module für die Visualisierung der aufbereiteten Daten in MeVisLab eingebunden. Die aus den Bildverarbeitungsvorgängen gewonnenen Daten werden auf Basis von Darstellungstechniken

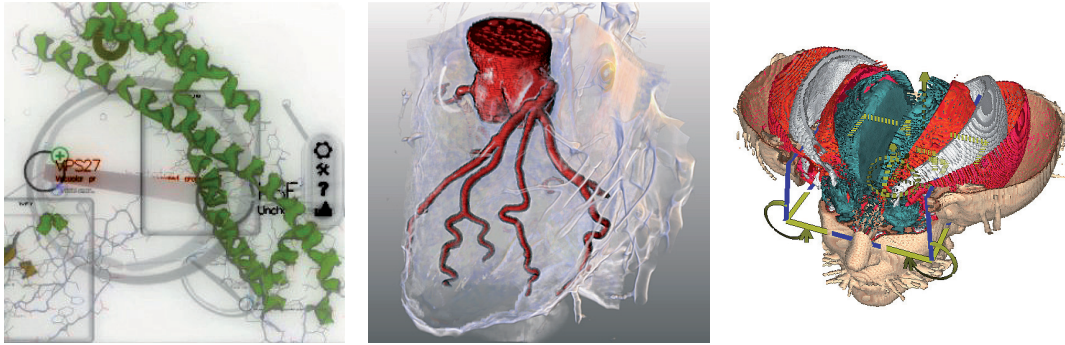


Abbildung 4: 3D-Benutzerschnittstellen für wissenschaftliche Visualisierungen. Die Benutzerschnittstelle der Anwendung INTERACTORIUM, aus [WIDJAJA U. A. 2009] (links), eine 3D-Visualisierung in der Software MeVisLab, aus [MEVISLAB] (Mitte) und eine interaktive Illustration im System von [MCGUFFIN U. A. 2003] (rechts).

aus dem Bereich des Non-Photorealistic Rendering für illustrative Darstellungen (beispielsweise Punktierung und Konturenverstärkung) der Untersuchungsgegenstände genutzt (siehe Abbildung 4, Mitte).

Eine weitere Anwendung für den medizinisch-wissenschaftlichen Bereich ist ein prototypische 3D-System, das von MCGUFFIN an der Universität von Montreal entwickelt wurde. In dieser Applikation werden medizinische Daten als computergrafische Voxel<sup>37</sup> dargestellt. Der Vorteil eines Voxelinsatzes zur Visualisierung besteht in der hohen Qualität der erzeugten Bilder und der Fähigkeit, die visuellen Strukturen ohne explizite Definition der Oberflächengeometrie zu erzeugen. Ferner liefern die medizinischen Messgeräte in vielen Fällen die Daten, die sich mit nur geringer Bearbeitung als Voxel auffassen lassen. Diese Eigenschaften der visuellen Struktur werden in der Anwendung genutzt, um dem Anwender durch interaktive Deformationsvorgänge (siehe Abbildung 4, rechts) spezifische Einblicke in die erhobenen Daten zu ermöglichen.

### 3.1.3.2 Benutzerschnittstellen zur Veranschaulichung abstrakter Daten

Interaktive 3D-Softwaresysteme zur Veranschaulichung abstrakter Daten sind wesentliche Werkzeuge im Umgang mit großen Datenmengen, weshalb im Bereich der Datenvisualisierung eine Reihe von Anwendungen existiert. Der wesentliche Unterschied zu wissenschaftlichen Visualisierungen besteht darin, dass bei der Informationsvisualisierung abstrakte Daten ohne Raumbezug die Grundlage der Visualisierung bilden (vgl. [CARD U. A. 1999, S. 7]). In der Folge wird für eine grafische Darstellung ein Bezugsraum definiert, in dem die Daten als visuelle Strukturen verortet werden. Die Abbildung 5 zeigt Bildschirmschnappschüsse der nachfolgend beschriebenen 3D-Anwendungen, die eine räumliche Verortung von Daten zeigen.

Forscher des Xerox PARC entwickelten im Jahr 1991 den *Information Visualizer*, der *Cone-* und *Cam-Tree* Darstellungen zur Visualisierung von hierarchischen Datenbeständen in Form räumlicher Baumstrukturen umfasst (vgl. [ROBERTSON U. A. 1991]). Im Unterschied zu flächigen Strukturen in 2D-Benutzeroberflächen werden mithilfe der Tiefendimension des

<sup>37</sup> Der Begriff Voxel ist abgeleitet von der Bezeichnung Volumenpixel und ist ein dreidimensionaler Bildpunkt der durch drei Koordinaten beschrieben wird: Länge, Breite, Tiefe

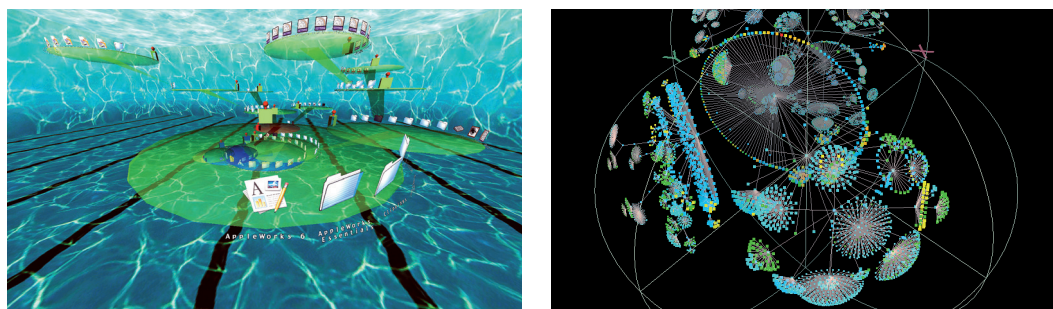


Abbildung 5: 3D-Interfaces im Bereich der Informationsvisualisierung. Die Darstellung einer Datenmenge als Cone-Tree in 3DosX, aus [CHIN 2002] (links) und die Benutzeroberfläche zur Darstellung von Graphen in der Anwendung Walrus, aus [CAIDA] (rechts)

Bezugsraums die Daten in hierarchischen, kegelförmigen Strukturen angeordnet. Durch eine halbtransparente Darstellung der Ordnungsstruktur wird eine gleichzeitige Sichtbarkeit der Daten unter Beibehaltung der Räumlichkeit ermöglicht. Diese Form der Visualisierung baumartiger Strukturen wird in dem Softwaresystem *3DosX*, einem 3D-Dateibrowser für das Betriebssystem Mac OS X, angewendet (vgl. [CHIN 2002, S. 16–17]). Die Ordnerhierarchie wird in der Umsetzung eines Cone-Tree durch miteinander verbundene kreisrunde Ebenen in einer zentralprojektiven Darstellung visualisiert. Auf den Ebenen sind die Ordner und Dateien gruppiert (siehe Abbildung 5, links). Dem Anwender werden für die Datenexploration festgelegte Interaktionsmöglichkeiten bereitgestellt, um einerseits zwischen den Dateien und Ordnern zu navigieren und andererseits mit den Daten zu operieren.

*Walrus* ist ein 3D-System zur interaktiven Visualisierung gerichteter Graphen. Die Entwicklung der Anwendung erfolgt als ein Open-Source-Projekt, das von der *Cooperative Association for Internet Data Analysis (CAIDA)* betreut wird (vgl. [CAIDA]). *Walrus* ist eine plattformunabhängige Software, die eine Visualisierung großer Datenmengen (circa 100.000 Elemente) ermöglicht. Die Verbildlichung der Daten erfolgt, indem die Elemente als Knoten in einer Graphenstruktur dargestellt werden (siehe Abbildung 5, rechts). Die Anordnung der Knoten im Graph und deren Visualisierung in einer zentralprojektiven Darstellung würde jedoch zu gegenseitigen Verdeckungen der Elemente im interaktiven Dialog mit dem Nutzer führen, wodurch die Verständlichkeit der Darstellung stark beeinträchtigt wäre. Für eine möglichst effiziente Darstellung der Graphen wird daher die Abbildungsfläche einer Kugeloberfläche nachempfunden (vgl. [MUNZNER 1997, S. 2]). Der Abbildungsvorgang wird ferner durch das Selektieren von wesentlichen Informationen in Abhängigkeit von der jeweiligen Betrachterposition unterstützt. Dazu wird die Festlegung des Spannbaums in der Anwendung vom Nutzer ausgeführt, wodurch situative Sichten auf die Datenmenge erzeugt werden können.

### 3.1.3.3 Benutzerschnittstellen zur Veranschaulichung virtueller Umgebungen

Virtuelle Umgebungen, die auf räumlichen Landschafts- und Gebäudemodellen sowie einer Menge an realitätsnachahmenden Objekten und Gegenständen basieren, sind die Grundlage für eine wachsende Zahl von kommerziellen und wissenschaftlichen 3D-Anwendungen im Kontext der Planung, Konstruktion, Simulation und Produktion.





Abbildung 6: 3D-Benutzerschnittstellen für Architektur- und Landschaftsvisualisierungen. Das Interface eines prototypischen Navigationssystems, aus [LORENZ U. A. 2008] (links) und ein Modell einer Badeinrichtung in einer CAD-Anwendung, aus [LUFT U. A. 2008] (rechts)

Im Fachbereich des Produktdesigns dienen CAD-Anwendungen zur Konstruktion, Bearbeitung und Visualisierung von Konsum- und Investitionsgütern. CAD-Systeme wie beispielsweise *ICEM Surf* [DASSAULT SYSTÈMES] oder *Autodesk Inventor* [AUTODESK] werden als Softwarewerkzeuge eingesetzt, um einerseits in den frühen Phasen einer Produktentwicklung Entwurfsvarianten, Skizzen und konzeptionelle 3D-Darstellungen zu erzeugen, andererseits im weiteren Projektverlauf präzise und detaillierte Konstruktionen zu erstellen. Dabei wird über den gesamten Entwurfsprozess hinweg eine Vielzahl von Daten erhoben, die als Metadaten an die zukünftigen Güter angefügt werden. Diese Informationen können in der CAD-Anwendung in verschiedenen Ansichten dargeboten werden, um letztlich eine möglichst reale und umfassende Darstellung eines zukünftigen Produktes darzubieten. Luft u. a. zeigen mit dem in [LUFT U. A. 2008] vorgestellten Framework, dass zusätzlich zu fotorealistischen Visualisierungsverfahren ebenso nicht-fotorealistische Rendertechniken in bestehende CAD-Systeme integriert werden können. Durch illustrative Darstellungen von Entwürfen und Modellen in CAD-Werkzeugen können einerseits konkrete Eigenschaften wie beispielsweise Form, Struktur und Materialbeschaffenheit von Objekten hervorgehoben (vgl. [GOOCH U. A. 1998]) oder aber spezifische Aussagen über ein Produkt kommuniziert werden (siehe Abbildung 6, rechts).

Im Kontext der Visualisierung virtueller Städte und Landschaften bieten Navigationssysteme für Fußgänger und Autofahrer, unter Berücksichtigung der Position des Nutzers und weiterer nutzerspezifischer Kriterien, mithilfe des 3D-Modells eine bildgestützte Zielführung zu einem gewählten Ort. Die experimentelle Navigationssoftware des Hasso-Plattner-Institutes (vgl. [LORENZ U. A. 2008]) bedient sich zur Darbietung der virtuellen Umgebung einer nichtlinearen Darstellungsweise um die flächige Landkartendarstellung mit einer räumlichen Sicht auf die Daten zu kombinieren (siehe Abbildung 6, links). In der Anwendung wird die unmittelbare virtuelle Umgebung des Nutzers auf Basis einer Zentralprojektion verbildlicht, um die Perspektive des Anwenders nachzubilden. Diese Darstellung wird mit einer flächigen Ansicht auf entfernte Bereiche des 3D-Modells kombiniert, wodurch eine Fokus-und-Kontext-Darstellung der komplexen virtuellen Umgebungen erreicht und die Orientierung des Nutzers begünstigt wird.

### 3.1.4 Nutzungskontext von 3D-Benutzerschnittstellen

Auf Basis von Evaluationen und Nutzerstudien werden seit vielen Jahren 3D-Benutzerschnittstellen analysiert und beurteilt sowie dessen zusätzlicher Nutzen für die Mensch-Computer-Interaktion diskutiert. Die wissenschaftlichen Arbeiten resümieren Vor- und Nachteile von 3D-Interfaces auf Basis verschiedener Daten und unter Einsatz unterschiedlicher Darstellungsweisen, was im Hinblick auf komponierte Abbildungsverfahren zur weiteren Forschung motiviert.

In den Arbeiten [TUFTE 2001, KAP. 7] und [WEN 1995, S. 1–2] stellen TUFTE und WEN heraus, dass für 2D-Daten eine Darstellung in einem planaren Bezugsraum ausreichend ist. Weiterhin wird konstatiert, dass ein höherdimensionaler Raum nur eingesetzt werden sollte, wenn dies zu einer semantischen Bereicherung der Darstellung führt. In [WISS U. A. 1998] wird zur Untersuchung dieses Kontextes eine vergleichende Studie zwischen den 3D-Visualisierungswerkzeugen *Cam-Tree* (vgl. [ROBERTSON U. A. 1991] und 3.1.3.2), *Information Cube* (vgl. [REKIMOTO & GREEN 1993]) und *Information Landscape* (vgl. [TESLER & STRASNICK 1992]) durchgeführt. Die Studie belegt, dass es zu inhärenten Abbildungsproblemen kommt, wenn Datensätze zugrunde gelegt werden, welche die Charakteristiken für eine räumliche Darstellung nicht aufweisen. Ferner existieren Studien, in denen 2D-Benutzeroberflächen den 3D-Schnittstellen hinsichtlich deren Gebrauchstauglichkeit gegenübergestellt werden, um Vor- und Nachteile von 3D-Interfaces zu identifizieren (vgl. [DACHSELT 2004, S. 38 FF.]). In Studien von ARK U. A. wurde bei Objektsuch- und Akquisitionsaufgaben eine Präferenz für 3D-Benutzerschnittstellen mit einer zentralprojektiven Abbildungsvorschrift gegenüber einer zweidimensionalen Darstellung aufgezeigt. Gleichzeitig wird jedoch von den Autoren darauf hingewiesen, dass die Studienergebnisse nicht verallgemeinerbar sind (vgl. [ARK U. A. 1998, S. 216 FF.]). Eine Evaluation, die von COCKBURN und MCKENZIE durchgeführt wurde, zeigt wiederum, dass ein erhöhter Freiheitsgrad in räumlichen Benutzerschnittstellen zu einer Leistungsverringerung führt, wenn Probanden die Schnittstelle als unübersichtlich und ineffizient empfinden [COCKBURN & MCKENZIE 2002, S. 209]. Ferner weisen die Autoren darauf hin, dass je nach konkreter Aufgabe und Anwendungsdomäne Argumente für einen Einsatz aber auch gegen den Gebrauch von 3D-Benutzerschnittstellen sprechen.

COCKBURN und MCKENZIE sowie ARK nutzen in ihren Vergleichsstudien jeweils zentralprojektive Abbildungsverfahren für die 3D-Benutzerschnittstelle, weshalb sich die Ergebnisse und Aussagen jeweils auf diese Darstellungsweise stützen. In neueren Studien wird darüber hinaus der Einsatz nicht-fotorealistischer Darstellungsweisen hinsichtlich der bildgestützten Vermittlung von Informationen sowie der Effektivität bei der Ausführung von Aufgaben mit 3D-Benutzerschnittstellen untersucht. In [ZAVESKY 2012] wird der Einsatz von multiperspektivischen Darstellungen für eine proportionswahrende und ausrichtungserhaltende Überführung von Objektraumelementen in den Bildraum untersucht. Die Studienergebnisse zeigen, dass die nicht-fotorealistische Darstellung anthropomorpher Formen zu einer erwartungskonformen Visualisierung der Objekte führen kann. In [ROSEN & POPESCU 2011, S. 629 FF.] dient die Bilderzeugung auf Basis mehrerer Kameras zur Realisierung von Überblicksdarstellungen einer komplexen visuellen Struktur in einer interaktiven 3D-Anwendung. In den Nutzerstudien werden Objektsuch- und Akquisitionsaufgaben betrachtet, die einerseits mittels einer multiperspektivischen Panoramadarstellung, andererseits mit einem zentralprojektiven Abbild

durchgeführt werden. Die Ergebnisse der Studie zeigen eine effizientere Bearbeitung der gestellten Aufgaben auf Basis der multiperspektivischen Darstellung.

Zusammenfassend soll eruiert sein, dass zum derzeitigen Stand der Forschung keine allgemeingültige Aussage zum Nutzungskontext und vor allem zum (bild)strukturellen Aufbau von 3D-Benutzerschnittstellen getroffen werden kann. Daher werden wesentliche Argumente für und gegen den Einsatz von 3D-Benutzerschnittstellen, aufbauend auf den Ausführungen in [MULLET & SCHIANO 1995], [SHNEIDERMAN 2003] und [DACHSELT 2004], aufgeführt und durch weitere Aspekte aus der Literatur ergänzt.

#### 3.1.4.1 Argumente für den Einsatz von 3D-Benutzerschnittstellen

- Die Visualisierung von Daten, die über eine inhärente dreidimensionale Struktur verfügen, können unter Beibehaltung der Dimensionalität im Objektraum verortet und deren relevanten Charakteristika im Bildraum intuitiv erfasst werden (vgl. [TUFT 2001; WEN 1995]).
- Durch eine Nachbildung der menschlichen Umwelt unter Beibehaltung des dreidimensionalen Eindrucks kann im Umgang mit 3D-Interfaces ein räumliches Orientierungs- und Handlungswissen abgeleitet werden. Dieses Wissen beruht primär auf der Leistungsfähigkeit des räumlichen Gedächtnisses der Nutzer (vgl. [SCHUMANN & MÜLLER 2000]).
- Zu den kognitiven Aspekten in der Mensch-Maschine-Kommunikation gehört die Aufmerksamkeitssteuerung. 3D-Benutzerschnittstellen erlauben eine vielfältigere visuelle Differenzierung durch visuelle Variablen (vgl. [MACKINLAY 1986]) als es durch 2D-Oberflächen möglich ist, da 3D-Interfaces die Aufmerksamkeit eines Anwenders durch die Verwendung zusätzlicher visueller Parameter lenken können (vgl. [DARKEN & SIBERT 1993]). Zu diesen zählen beispielsweise die Lagebeziehungen im Raum oder die relative Position des Nutzers zum Horizont (vgl. [GROH 2008]).
- Der höherdimensionale Informationsraum und der höhere Freiheitsgrad von 3D-Benutzerschnittstellen eignen sich im Besonderen für die Präsentation mehrdimensionaler Datenmengen (vgl. [SCHÖNHAGE 2001]), da die zusätzliche Raumkoordinate und die Objektausrichtung für die Interfacegestaltung genutzt werden können (vgl. [DÄBLER & PALM 1998]). Dadurch ist eine verbesserte Informationsdichte im Vergleich zu 2D-Benutzerschnittstellen möglich (vgl. [ROBERTSON U. A. 1993]).

#### 3.1.4.2 Argumente gegen den Einsatz von 3D-Benutzerschnittstellen

- Die Orientierung in 3D-Benutzerschnittstellen ist eine wesentliche Herausforderung in interaktiven Anwendungen, weil diese durch fehlende Bezugspunkte und Verhältnisse innerhalb des Informationsraums gestört sein kann [NIELSEN 1998]). Die räumlichen Benutzerschnittstellen erfordern daher vom Nutzer bei Interaktions- und Navigationsaufgaben einen hohen kognitiven Aufwand (vgl. [MACKINLAY U. A. 1990]) und erschweren die Fokussierung auf die tatsächliche Arbeitsaufgabe (vgl. [SHNEIDERMAN 2003]).

- In räumlichen Darstellungen mit wenigen, respektive nicht ausreichend vielen räumlichen Tiefenhinweisen kann es zu Mehrdeutigkeiten bei der Positions-, Größen- und Ausrichtungsbestimmung von Objekten kommen (vgl. [SMALLMAN U. A. 2001; SCHUMANN & MÜLLER 2000]). Dies beeinträchtigt die Informationsgewinnung und darüber hinaus die Einschätzung von Entfernungen [DÄBLER 1999] bis zum Grad einer verfälschten Informationsdarbietung [TORY U. A. 2004].
- Der Anwender erfährt bei räumlichen Darstellungen einen Informationsverlust, wenn entfernte Objekte durch näherliegende Elemente verdeckt werden (vgl. [CARD U. A. 1999; DÄBLER 1999]).
- Im Gegensatz zu fensterbasierten 2D-Benutzerschnittstellen existiert für den Entwurf von räumlichen Benutzerschnittstellen bisher keine etablierte Metapher, wodurch die Entwicklung von 3D-Benutzerschnittstellen erschwert wird (vgl. [GROH 2007; BOWMAN U. A. 2004]).
- Trotz der Entwicklungen im Grafikbereich sind 3D-Benutzerschnittstellen durch einen deutlich höheren Ressourcenverbrauch als 2D-Interfaces und können infolgedessen durch inakzeptable Bildwiederholungsraten gekennzeichnet sein. Einige Ausprägungen von 3D-Interfaces können dadurch nur eingeschränkt in interaktiven Anwendungen verwendet werden (vgl. [AKENINE-MÖLLER U. A. 2008]).
- Bei der Integration von per se flächigem Text (Worte) in dreidimensionalen Darstellungen vermindert sich dessen Lesbarkeit mit einem zunehmenden Abstand zum Betrachter sowie bei einer Ausrichtung der Worte in die Raumtiefe. Dadurch wird das Verständnis für die zu vermittelnden Informationen beeinträchtigt (vgl. [SHNEIDERMAN 2003]). Eine Darstellung des Textes als Fläche verbindet sich indes nicht mit der Räumlichkeit einer perspektivischen Darstellung (vgl. [GROH 2008, S. 3]).

### 3.1.5 Notwendigkeit einer systematischen Betrachtung

Aufbauend auf der Identifikation von Vor- und Nachteilen von 3D-Benutzerschnittstellen wird resümierend auf die Notwendigkeit einer Betrachtung des Nutzer-Bild-Dialogs in interaktiven Anwendungen hingewiesen. Anhand verschiedener Auszüge aus der Literatur der letzten Dekade wird eine systematische Betrachtung von 3D-Interfaces motiviert.

In verschiedenen wissenschaftlichen Arbeiten wird ein erheblicher Forschungsbedarf auf dem Gebiet des Interfacedesigns interaktiver 3D-Anwendungen identifiziert (vgl. [CHEN 2005, S. 13; JOHNSON 2004, S. 14,16]). BOWMAN kritisiert in diesem Zusammenhang das zu kleine Repertoire an kontextsensitiven 3D-Benutzerschnittstellen, welches sich in der letzten Dekade weder quantitativ noch qualitativ weiterentwickelt hat (vgl. [BOWMAN U. A. 2006, S. 5]). Ferner identifiziert BOWMAN dies als eine Ursache für die Vielzahl von weiterhin bestehenden Forschungsfragen und den fehlenden Zugewinn an konstruktiven 3D-Interfaces im betrachteten Zeitraum (vgl. [BOWMAN U. A. 2006, S. 6]). Dabei weist der Forschungsbereich computergrafischer Visualisierungen eine breite Basis an informellem Wissen im Hinblick auf die Gestaltung und Anwendung von 3D-Benutzerschnittstellen auf. Diese mündet jedoch nur unsystematisch und zu eingeschränkt in konkreten Entwicklungen [BOWMAN U. A. 2008,

S. 32]. SHNEIDERMAN führt die fehlenden Entwicklungen vor allem auf die eigens auferlegte Restriktion einer zu begrenzt verstandenen Eins-zu-Eins-Abbildung der realen Welt auf die virtuelle Umgebung zurück (vgl. [SHNEIDERMAN 2003, S. 12]). Ferner zeigen die bisherigen Betrachtungen, dass ein unflexibler und starr eingesetzter Bilderzeugungsvorgang mitverantwortlich für die bestehenden Defizite ist. Die Bereitstellung von zusätzlichen Freiheitsgraden in der Gestaltung von 3D-Benutzerschnittstellen wird in [DURAND 2002A, S. 111–112] und in [DEUSSEN 2001, S. 79] gefordert. Die Einbindung flexibler Darstellungstechniken in die Gestaltung interaktiver 3D-Systeme wird beispielsweise in [BEHRENS U. A. 2010], [BÖTTGER U. A. 2008] und [LORENZ U. A. 2008] aufgezeigt. STUERZLINGER und WINGRAVE stellen jedoch heraus, dass zusätzlich zur allgemeinen Gestaltungsvielfalt und der konkreten Gestaltung einer Benutzerschnittstelle, die generelle Anwendbarkeit von Darstellungstechniken zu prüfen ist. Dazu ist es notwendig die Charakteristiken einer Benutzerschnittstelle unter allgemeingültigen Gesichtspunkten zu identifizieren (vgl. [BOWMAN U. A. 2006, S. 6–7]) und Gestaltungshinweise abzuleiten (vgl. [STUERZLINGER & WINGRAVE 2011, S. 211 ff.]). In der Konsequenz wirken die bisher unzureichende Systematisierung und das fehlende Aufzeigen von Möglichkeiten und Anwendungskontexten, der Nutzung und der Verbreitung von neuen Darstellungstechniken entgegen. Dieses Fehlen kann nicht allein durch Visualisierungsbibliotheken und 3D-Frameworks, beispielsweise *ParaView* [CEDILNIK U. A. 2006]), kompensiert werden, da die darauf aufbauenden Visualisierungssysteme die Darstellungstechniken lediglich bereitstellen und ferner nur in vordefinierten Anwendungsgebieten verfügbar machen.

Auch wenn bestehende Entwicklungs- und Entwurfswerkzeuge den Realisierungsprozess unterstützen, müssen die 3D-Anwendungen aufwendig und individuell implementiert werden (vgl. [TAKALA U. A. 2012]). Dies offenbart einen Forschungs- und Entwicklungsbedarf, um 3D-Benutzerschnittstellen effizient und effektiv in domänenspezifischen, interaktiven Anwendungen zur Verfügung zu stellen. Dabei ist ein grundlegendes Verständnis für den bildgestützten Interaktionsprozess zwischen Nutzer und System ein notwendiger Schritt. Das etablierte Wissen aus den Bereichen des Interface- und des Interaktionsdesigns muss in der 3D-Interfaceentwicklung berücksichtigt werden, um die wesentlichen Herausforderungen im Nutzer-Bild-Dialog zu lösen (vgl. [CHEN 2005; BOWMAN U. A. 2012, S. 86–87]). Die aufgezeigten Anwendungen (siehe Unterabschnitt 3.1.3) und die vorgestellten Ausprägungen computergrafischer Darstellungen (siehe Unterabschnitt 2.1.2) zeigen, dass Darstellungsweisen abseits des computergrafischen Fotorealismus Gestaltungsmöglichkeiten für effiziente und effektive 3D-Benutzerschnittstellen sein können. Darstellungen, die durch einen veränderten Projektionsvorgang erreicht werden, sind im Anwendungskontext jedoch kaum berücksichtigt. Unkonventionelle Projektionsverfahren gehören zu den weniger erforschten Bereichen und weisen lediglich experimentelle Bezüge zu einzelnen Anwendungsbereichen auf. Der Einsatz von einem erweiterten Repertoire an Darstellungstechniken in interaktiven 3D-Anwendungen wird ferner durch Studien von ROSEN und POPESCU motiviert. Die Autoren plädieren für einen Bilderzeugungsprozess *„that abandons the traditional rigidity of the camera model in favor of designing the camera according to the application needs, and optimizing it dynamically“*<sup>38</sup> [ROSEN & POPESCU 2011, S. 631].

<sup>38</sup> „[Ein Bilderzeugungsprozess] welcher die traditionelle Steifigkeit des Kameramodells aufbricht und durch ein Modell, das Anwendungsanforderungen und eine dynamische Optimierung berücksichtigt, ersetzt“



Eine Vielzahl von Forschungs- und Entwicklungsarbeiten wird notwendig sein, um generalisierte computergrafische Darstellungen in allgemeingültigen beziehungsweise standardisierten Verfahrensweisen für interaktive 3D-Anwendungen zu etablieren. Eine weitergehende Betrachtung linearer und nichtlinearer Darstellungsformen abseits des computergrafischen Abbildungsprozesses soll zunächst Lösungsansätze für bestehende Herausforderungen aufzeigen und eine Basis für die weitergehende Forschung und Entwicklung räumlicher Benutzerschnittstellen bieten.

## 3.2 Formen und Eigenschaften von 3D-Darstellungen

Der strukturelle Aufbau eines Bildes beziehungsweise eines Abbildes wird durch die Gestalt und die Ordnung der Elemente sichtbar. Dadurch werden Aussagen über die zugrunde liegende Primärgeometrie sowie die Verfahren und die Techniken, die zur Erstellung eines Bildes angewendet wurden, möglich. Prinzipiell kann der Ausgangspunkt für eine systematische Betrachtung folglich der Objektraum oder der Bildraum sein. Die Techniken, die zur Erstellung eines Bildraums eingesetzt werden, können jedoch nur in Einzelfällen durch eine Analyse und Rekonstruktion des Objektraumes bestimmt werden. Die Ursache dessen besteht in der fehlenden Gewissheit hinsichtlich der tatsächlichen Szenerie, die als Grundlage diene. Die Festlegung, eine Analyse aus dem Bildraum heraus zu führen, wird darüber hinaus aus dem Aspekt heraus substantiiert, dass die Bildaussagen in die Betrachtungen einbezogen werden können. Die bildstrukturellen Betrachtungen können hierdurch ergänzt werden. DURAND verdeutlicht den Nutzen einer bildraumbasierten Analyse exemplarisch. Er weist darauf hin, dass sich im Bildraum der Unterschied und die Bildwirkung zwischen Ein-, Zwei- und Drei-Fluchtpunktperspektive erklären lassen, obwohl der jeweilige Bildraum auf einer identischen geometrischen Struktur des Objektraums basiert (vgl. [DURAND 2002A, S. 113]). In der Konsequenz ist eine Betrachtung ausgehend vom Bildraum mit Bezügen zum Objektraum, indem Charakteristiken bestehender Projektionstechniken aufgezeigt werden, zweckdienlich. Eine in solcher Weise aus dem Bildraum intendierte Betrachtung kann in mannigfaltiger Form erfolgen. Daher werden zunächst systematische Betrachtungen von 3D-Darstellungen aus unterschiedlichen Wissenschaftsbereichen kurz vorgestellt:

- In den Arbeiten von FOLEY [FOLEY 1995] und SALOMON [SALOMON 2006], zu den Grundlagen der Computergrafik, werden Systematisierungen von computergrafischen Projektionen erläutert. Hierbei werden, auf Basis mathematischer Definitionen und Beschreibungen, Projektionsvorschriften vorgestellt, erläutert und anhand von Beispielabbildern verdeutlicht. Aufgrund des fehlenden Bezugs zu bildintentionalen Aspekten werden diese Arbeiten nicht als Grundlage der Betrachtungen dienen, jedoch die Bezüge zur Computergrafik hinsichtlich der technischen Beschreibung von Projektionen sicherstellen.
- In WALTHERS „Malerei der Welt“ [WALTHER 1998] wird die Entwicklung der abendländischen Malerei vom Mittelalter bis zur Gegenwart dargestellt. Über 900 Werke werden vor dem jeweiligen zeitgeschichtlichen Hintergrund betrachtet. Die Arbeit beleuchtet eine sehr große Anzahl an künstlerischen Werken und ordnet diese vorrangig zeitlich

und innerhalb jeder Kunstepoche nach Künstlern. WALTHER bietet damit keine Klassifikation nach bildstrukturellen Eigenschaften oder Darstellungsintentionen, sondern eine Ordnung von Werken aus unterschiedlichen Epochen, im Hinblick auf die Philosophien und die Techniken der Künstler vor dem jeweiligen zeitgeschichtlichen Hintergrund. Die Bildanalysen WALTHERS bieten damit einen Fundus künstlerischer Werke für Betrachtungen abseits der Computergrafik und darüber hinaus Interpretationen zu Motiven und Hintergründen der Künstler in deren jeweiliger Zeit.

- In [HOCKNEY 2006] werden über einhundert Werke aus verschiedenen Kunstepochen analysiert und einander gegenübergestellt, um Entwicklungen und Veränderungen in der Malweise von Gesichtern, Gewändern und Perspektiven aufzuzeigen. HOCKNEY erläutert in seiner Arbeit, dass die europäische Malerei seit dem frühen 15. Jahrhundert ohne fundierte Kenntnisse über optische Abbildungstechniken nicht zu verstehen ist. Zur Verdeutlichung der Aussagen wird eine Vielzahl von Bildanalysen durchgeführt, die den Erstellungsprozess und die dabei genutzten Projektionsinstrumente beschreiben. Darüber hinaus werden Bezüge zur Gegenwart hergestellt, indem die Möglichkeiten der Künstler im Computerzeitalter hinterfragt und Impulse für das moderne Kunstschaffen gegeben werden. Die Einblicke HOCKNEYS dienen der Analyse von strukturellen Eigenschaften eines Bildraums und den verwendeten Abbildungsmethoden, bieten jedoch keine Systematisierung von Darstellungsformen.
- WILLATS analysiert in [WILLATS 1997] Abbildungsmethoden anhand von Studien zu Bildstrukturen. WILLATS verknüpft in seiner Analyse die bildstrukturellen Eigenschaften von Werken mit den vermutlichen Darstellungsintentionen der Künstler. Dabei betrachtet er die Eigenschaften räumlicher Szenerien einerseits und die Charakteristika des Bildes andererseits, um herauszustellen, durch welches Verfahren die Repräsentationen visueller Strukturen erzeugt wurden. Weiterhin überführt WILLATS die gewonnenen Erkenntnisse in eine Klassifikation von Abbildungsverfahren. Obwohl nichtlineare Darstellungsmethoden in der Arbeit betrachtet werden, erfolgt keine generelle Klassifikation, sondern lediglich eine Systematisierung linearer Projektionsverfahren.
- Die Ausführungen WILLATS sind der Ausgangspunkt für eine Klassifikation linearer und nichtlinearer Darstellungsmethoden in [DURAND 2001]. Vergleichbar mit den Kriterien in [WILLATS 1997] bilden in der Arbeit von DURAND ebenfalls die bildstrukturellen Eigenschaften und die resultierenden Bildintentionen die Grundlage für dessen Systematisierung. In dieser erfolgt die Einteilung von Darstellungsmethoden unabhängig von der Kunstepoche oder dem Erzeugungsmedium, die in der Folge eine vielfältige Betrachtung zulässt und eine Überführung der Erkenntnisse in computergrafische Darstellungstechniken bietet.

Die Analysen und Erläuterungen dieser Arbeiten zeigen, dass abseits der Ähnlichkeitsparadigmen und des Naturalismus in den Epochen vor der Fotografie und der Computergrafik Bilder entstanden, die eine intentionale Bildgestaltung einsetzten, um Botschaften und Aussagen gezielt transportieren und kommunizieren können. Vor allem die Arbeiten von HOCKNEY, WILLATS und DURAND veranschaulichen, dass sich die Entwicklungen in der neuzeitlichen Kunst seit der Renaissance im Spannungsfeld zwischen einem mathematisch korrekten

Abbildungsprozess, der Berücksichtigung der menschlichen visuellen Wahrnehmung und der intentionalen Bildgestaltung befindet. Insbesondere bei der Betrachtung von Werken der neuzeitlichen Malerei wird deutlich, dass die Künstler ähnlichen Herausforderungen gegenüberstanden, wie die Entwickler und die Designer derzeitiger computergrafischer 3D-Darstellungen: die Abbildung eines dreidimensionalen Raumes auf eine zweidimensionale Fläche, indem die Gesetze der Optik und der Geometrie methodisch angewendet werden (vgl. [GROH 2007, S. 24 ff.]). Hieraus resultieren unterschiedliche Darstellungstechniken zur Erzeugung von Räumlichkeit, in denen sich die Sichtweise des Künstlers auf dessen Umgebung ausdrückt. Infolgedessen können sich vor historischen und theoretischen Hintergründen der neuzeitlichen Malerei alte Techniken als Innovation für die gegenwärtige Computergrafik erweisen. Die bildenden Künste zeigen sich dabei in zweierlei Hinsicht als Wegbereiter einer Darstellungsvielfalt in computergrafischen Visualisierungen. Die Werke zeigen einerseits formale Kontextfaktoren, die in den Erstellungsprozess einfließen und gleichsam Darstellungszwecke für einen differenzierten Einsatz von Visualisierungen. Eine vollständige Aufarbeitung des historischen Hintergrundes sowie der Geisteshaltung, die zur Schaffung der nachfolgend betrachteten Werke führte, ist im Rahmen dieser Arbeit einerseits nicht zielführend und wird andererseits von einer Vielzahl von Fachleuten bereits vollzogen. Daher sei an dieser Stelle auf die kunstgeschichtliche Fachliteratur<sup>39</sup>, auf Arbeiten der Kunsthistoriker PANOFKY und FLORENSKI<sup>40</sup> und auf die Literatur im Bereich der Bildwissenschaft<sup>41</sup> verwiesen.

Ein Teil der Systematisierung aus [DURAND 2001] fließt in die nachfolgende Betrachtung ein, die jedoch gegenüber den Ausführungen von DURAND im Hinblick auf die Überführung von Darstellungstechniken in computergrafische Visualisierungsverfahren konkretisiert und um zusätzliche Aspekte erweitert ist. Die nachfolgenden Analysen beziehen sich auf die perspektivische Struktur von Bildern sowie deren Eigenschaften und stützen sich auf Erkenntnisse aus dem bildsprachlichen sowie dem computergrafischen Bereich. Als Vertreter der *Visualistik*<sup>42</sup>, die zur Bildung eines hierbei notwendigen Begriffsapparates beigetragen haben, sind vor allem PANOFKY durch dessen Analysen der Raumdarstellung des Mittelalters und der Neuzeit, ARNHEIM in Anbetracht von dessen Kompositionslehre (vgl. [ARNHEIM 2003]) und GROH aufgrund der Perspektivanalysen künstlerischer Werke (vgl. [GROH 2007]) zu nennen. Weiterhin sind HOCKNEY infolge seiner Analyse von Abbildungstechniken in [HOCKNEY 2006] und WILLATS durch die Betrachtung von Darstellungsweisen (vgl. [WILLATS 1997]) anzuführen. Die Begriffe aus dem Bereich der Bildsprache werden durch Definitionen und Festlegungen aus der Computergrafik (u.a. [FOLEY 1995; ANGEL 2005; AKENINE-MÖLLER U. A. 2008; SALOMON 2006]) ergänzt, sodass ein dem Verständnis des Kontextes zweckdienlich zur Verfügung stehender Begriffsraum existiert.

<sup>39</sup> siehe [GOMBRICH 1996]

<sup>40</sup> siehe [PANOFKY 1980] und [FLORENSKI 1997]

<sup>41</sup> siehe [GRAU 2003] und [BÜTTNER 2005]

<sup>42</sup> Der Fokus der interdisziplinären Bildwissenschaft liegt nicht ausschließlich auf Bildern oder Kunstwerken; dieser richtet sich vielmehr auf die menschliche Fähigkeit, Bilder gezielt zu erzeugen und als Kommunikationsmedium einzusetzen, um diese wahrnehmen (rezipieren) und verwenden zu können (vgl. [SCHIRRA 2006]).



### 3.2.1 Lineare Darstellungsformen

Ein Abbild auf Basis eines linearen Abbildungsverfahrens ist durch eine uniforme Bildstruktur und die Erhaltung der Geradlinigkeit charakterisiert (vgl. [FOLEY 1995, S. 280]). Die unter dem Begriff der *linearen Darstellungsformen* subsumierten Abbilder basieren auf perspektivischen und parallelen Projektionen (vgl. [SALOMON 2006, S. 2]). Die zwei Gruppen von Abbildungsverfahren unterscheiden sich in der Charakteristik des *Zentrums der Projektion*, da beide Projektionsarten über ein Zentrum der Projektion, in jedoch unterschiedlicher Lage, verfügen.

Die *zentralprojektive Darstellung* basiert auf dem Ansatz, dass der Objektraum ein unendliches, dreidimensionales Konstrukt ist, das durch ein ruhiges und unbewegtes Auge betrachtet wird. In der Folge ist das Zentrum der Projektion – an deren Position sich das Auge des Betrachters befindet – ein spezifizierbarer Punkt im Objektraum. Die Bewahrung der scheinbaren Räumlichkeit, in einem *monoperspektivischen Abbild*, entsteht durch eine strukturerhaltende Projektion und die gleichförmige Ordnung der Objekte im Bildraum. Die Abbildung 7-A veranschaulicht, dass durch die Projektionseigenschaften die Illusion von einem dreidimensionalen Raum erzeugt – eine Raumtiefe suggeriert – wird. Der räumliche Eindruck wird darüber hinaus gestärkt, da parallele Objektkanten im Bildraum, als in einem gemeinsamen Fluchtpunkt konvergierend, dargestellt werden [EYSENCK & KEANE 2010, S. 68 ff.]. Die resultierende perspektivische Verkürzung der Objekte bewirkt, dass deckungsgleiche Elemente des Objektraumes mit zunehmendem Abstand des Betrachters im Bildraum verkleinert abgebildet werden. Wenn in einer Primärgeometrie raumtiefenparallele Objektkanten existieren, vereinigen sich diese im Bild als Geraden in der optischen Verlängerung in einem Punkt auf der *Horizontlinie*. Dieser Punkt beschreibt den Durchstoßpunkt des *Haupt-sichtstrahls* auf der Bildebene, der sich im Bildraum im Schnittpunkt von *Horizontal-* und *Sagittallinie* befindet. Dieser Punkt wird in [ARNHEIM 2003] als *Hauptpunkt* und in [GROH 2007] als *geometrische Mitte* des Bildes bezeichnet und in Abbildung 7-A anhand einer geometrischen Struktur verdeutlicht. Die raumtiefenparallelen Kanten der Objekte verlaufen in der Verlängerung in den Punkt [M]; die geometrische Mitte beziehungsweise den Hauptpunkt.

Die parallelprojektive Darstellung basiert ebenfalls auf einem linearen Abbildungsverfahren, der *Parallelprojektion*, die einen Grenzfall der Zentralprojektion darstellt. Im Unterschied zum zentralprojektiven Abbild konvergieren die in die Tiefe des Objektraums führenden Kanten im Bildraum nicht, sondern verlaufen parallel in die Bildtiefe. In der Folge besitzen parallelprojektive Darstellungen keine geometrische Mitte. Ebenso wenig erfolgt bei zunehmender Tiefe im Objektraum eine perspektivische Verkürzung der Objekte im Bildraum. Durch die dargestellten Hilfslinien in Abbildung 7-B wird der strukturelle Aufbau einer parallelprojektiven Darstellung hervorgehoben und die Unterschiede zur Zentralprojektion aufgezeigt. Obwohl einige monokulare Tiefenhinweise (vgl. [GOLDSTEIN 2007, S. 185 ff.]) im parallelprojektiven Bildraum fehlen, entsteht in den Abbildern ebenfalls ein räumlicher Eindruck. Durch die Verdeckung von Objekten wird die räumliche Tiefe suggeriert, wenngleich die perspektivische Verkürzung und das Lageverhältnis von Objekten zum Horizont als Indikatoren fehlen.

Eine dritte lineare Darstellungsform ist die *invers-zentralprojektive Darstellung*. Im Vergleich zum zentralprojektiven Abbild weist der Bildraum invertierte bildstrukturelle Eigenschaften

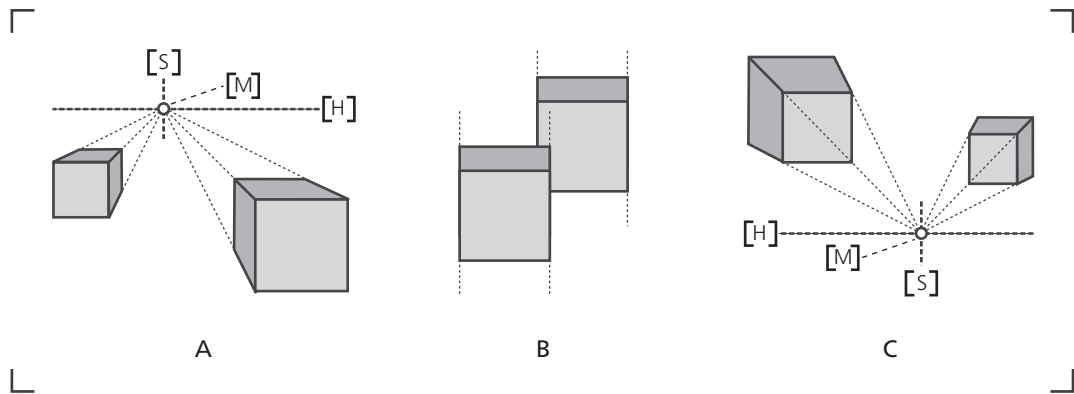


Abbildung 7: Schematische Darstellung linearer Darstellungsformen mit deren bildstrukturellen Eigenschaften und dem Hauptpunkt [M] (sofern vorhanden) im Schnittpunkt von Horizontlinie [H] und Sagittallinie [S] eines zentralprojektiven Abbildes (A), eines parallelprojektiven Abbildes (B) und eines invers-zentralprojektiven Abbildes (C)

auf. Die raumtiefenparallelen Kanten verlaufen optisch nicht in die Tiefe des Raumes, sondern besitzen einen entgegengesetzten Verlauf und divergieren folglich als Bildgeraden. Dadurch ist der Hauptpunkt scheinbar vor der Bildebene – beim Betrachter – verortet. Die Abbildung 7-C zeigt die bildstrukturellen Eigenschaften invers-zentralprojektiven Abbildes sowie die Position des Hauptpunktes [M]. Darüber hinaus wird deutlich, dass die im Bild auftretenden perspektivischen Verkürzungen bei dieser Form der linearen Darstellung ebenfalls invertiert sind.

### 3.2.2 Nichtlineare Darstellungsformen

Darstellungsformen, die keine einheitlich-linearen bildstrukturellen Eigenschaften aufweisen, werden in [DURAND 2001] mit dem Begriff „non-linear drawing system“<sup>43</sup> bezeichnet. Diese Benennung ist konform zur computergrafischen Fachliteratur (vgl. [FOLEY 1995, S. 237; SALOMON 2006, S. 2]), jedoch unzureichend für eine genaue Betrachtung der Eigenschaften der unter diesem Begriff subsumierten Darstellungsformen. Weder im Bereich der Bildsprache noch in der Computergrafik hat sich aufgrund der vielfältigen Ausprägungen bisher ein allgemeingültiger Begriff etabliert, welcher nicht ausschließlich die Negation der zuvor aufgezeigten linearen Abbilder fixiert. Die Ursache dieser unspezifischen Begriffsbildung verdeutlicht Salomon mit der Aussage: „*It seems that the number of possible nonlinear projections is vast and is limited only by the imaginations of those who try to develop new ones.*“<sup>44</sup> [SALOMON 2006, S. 145]. Während in der Literatur oftmals eine exemplarische Aufzählung nichtlinearer Darstellungsverfahren als Beschreibung dient (vgl. [FOLEY 1995, S. 237]), führt DURAND in [DURAND 2001] eine Ordnung von Darstellungsformen auf Basis bildstruktureller Eigenschaften an. In den folgenden Unterabschnitten dienen die Ausführungen DURANDS als Ausgangspunkt für die weiterführende Systematisierung.

<sup>43</sup> „Nichtlineares Abbildungssystem“

<sup>44</sup> „Es scheint, dass die Anzahl möglicher nichtlinearer Projektion überwältigend ist und nur durch die Vorstellungskraft derer begrenzt ist, die neue Projektionen entwickeln.“

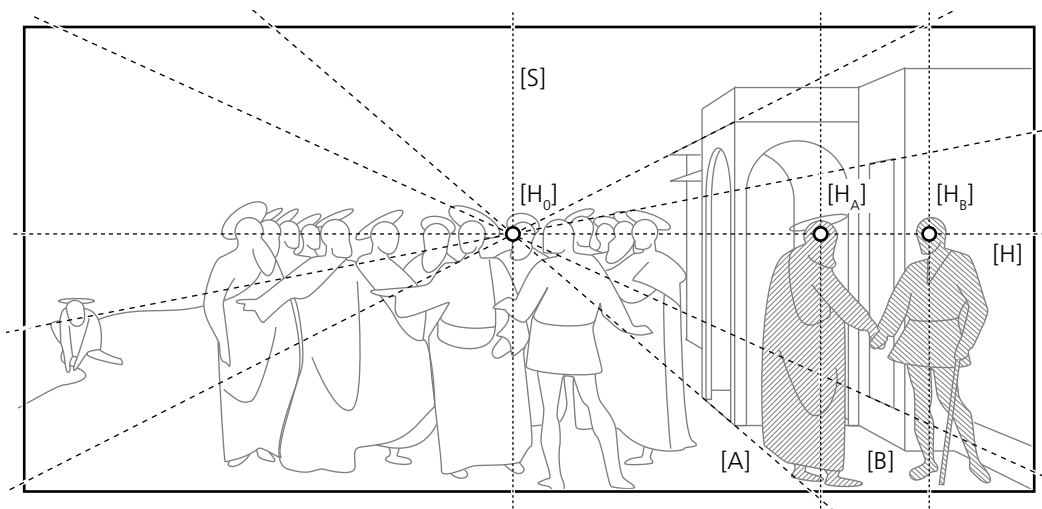
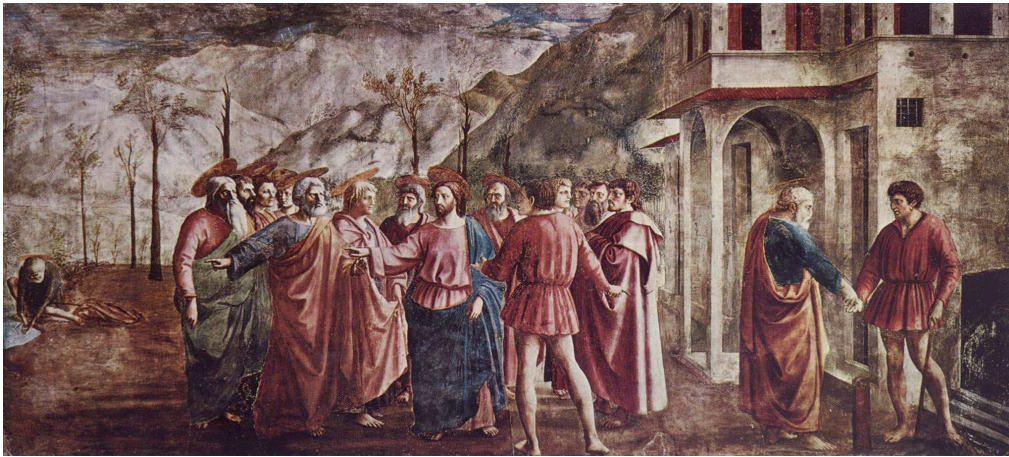


Abbildung 8: Objektraumbasierte Multiperspektive im Bild „Der Zinsgroschen“, TOMMASO DI SERCASSAI (MASACCIO), 1426-1427, aus [VON EINEM 1967, S. 83] (oben) und Skizze des Gemäldes, nach [WOJZIAK U. A. 2011E] mit Horizontlinie [H] und Sagittallinie [S] sowie dem Hauptpunkt  $[H_0]$  des Bildes und den zusätzlichen Hauptpunkten  $[H_A]$  und  $[H_B]$  der Figuren [A] und [B] (unten)

### 3.2.2.1 Multiperspektivische Darstellungen

*Multiperspektivische Bilder* entstehen im Gegensatz zu monoperspektivischen Abbildern (siehe Unterabschnitt 3.2.1) durch die Zusammenführung mehrerer perspektivischer Abbildern mit individueller uniform-linearer Bildstruktur. Ein Abbild umfasst jeweils einzelne Objekte oder Teilbereiche einer Szene und wird als Teil einer Komposition mit weiteren Abbildern in einem Bild zusammengeführt. Durch diesen Kompositionsvorgang erfolgt „die Vereinigung mehrerer Blickwinkel in einem Bild“ [FRANKE U. A. 2005, S. 1]). Die Abgrenzung der einzelnen Perspektiven basiert entweder auf den Elementen des Objektraumes oder vollzieht sich auf der Grundlage des Bildraumes. Die zwei Ausprägungen werden nachfolgend näher erläutert und begrifflich in »objektraumbasierte Multiperspektive« und »bildraumbasierte Multiperspektive« unterschieden.

## Die objektraumbasierte Multiperspektive

Die bildstrukturellen Eigenschaften einer objektraumbasierten multiperspektivischen Darstellung werden durch die Strukturanalyse des Gemäldes „Der Zinsgroschen“ (Abbildung 8) von TOMMASO DI SERCASSAI (1401-1428, genannt MASACCIO) verdeutlicht. Die Skizze des in Abbildung 8 zeigt, dass MASACCIOs Gemälde auf einer Komposition mehrerer zentralprojektiver Abbilder beruht. Zur Verifizierung der Darstellungsform werden die Hauptpunkte des Bildes identifiziert und die Struktur der Bildelemente betrachtet. Die Analyse offenbart eine Unterordnung der Architektur im rechten Bildbereich unter dem Hauptpunkt des Bildes [ $H_0$ ]. Dieser befindet sich im Schnittpunkt von Horizontal- [ $H$ ] und Sagittallinie [ $S$ ] und zeigt, dass der strukturelle Aufbau des Bildraums durch ein zentralprojektives Abbildungsverfahren erreicht wird. Im Unterschied zur Abbildung des Gebäudes sind die dargestellten Personen dieser geometrischen Struktur nicht unterworfen. Es scheint als habe sich der Künstler jeder einzelnen Figur beim Malen gesondert zugewandt und diese als separate Teildarstellungen in das Bild eingefügt (vgl. [HOCKNEY 2006]). Die unterschiedlichen Augpunkte des Künstlers, die sich durch das Zuwenden des Malers zu den einzelnen Personen ergeben, sind identisch positioniert. Lediglich die Ausrichtungen und damit der wandernde Blick des Künstlers induzieren die verschiedenen Perspektiven im Bild. Eine Bildanalyse zeigt, dass jede Person im Bild von TOMMASO DI SERCASSAI in einer individuellen Perspektive verbildlicht ist und in der Folge über einen eigenen Hauptpunkt verfügt (vgl. [WOJDAK U. A. 2011E]). Exemplarisch sind in Abbildung 8 die Hauptpunkte [ $H_A$ ] und [ $H_B$ ] der Personen [A] und [B] eingezeichnet. Durch die Integration zusätzlicher Perspektiven wandelt sich das monoperspektivische Abbild zu einem kompositorischen, multiperspektivischen Bild.

Der Einsatz individueller Perspektiven ermöglicht die Hervorhebung ausgewählter Objekte – die Personen in Masaccios „Der Zinsgroschen“ – oder Bereiche einer Szenerie. Nach GROH waren die separat abgebildeten Elemente für den Künstler meist von besonderer narrativer Bedeutung (vgl. [GROH 2007, S. 34]). Gleichzeitig werden durch die identischen Augpunktpositionen des Künstlers die Relationen der Objekte im Bildraum beibehalten. Darüber hinaus werden in [WARE 2009, S. 168–169] und in [GROH 2007, S. 25] aufgezeigt, dass durch die Zuwendung des Künstlers und die damit verbundene Verlagerung des Hauptpunktes projektionsbedingte Verzerrungen minimiert werden. In der Folge wird eine der visuellen Wahrnehmung des Menschen genügende Darstellung erreicht (vgl. [FRANKE U. A. 2008, S. 11-12; WOJDAK U. A. 2011E, S. 2–3]).

Eine weitere Ausprägung objektraumbasierter Multiperspektive kann anhand des Gemäldes „Stilleben mit Obstkorb“ (Abbildung 9) von PAUL CÉZANNE (1839-1906) aufgezeigt werden. Im Unterschied zum bildstrukturellen Aufbau von MASACCIOs „Der Zinsgroschen“ wird der Objektraum nicht nur aus unterschiedlichen Blickrichtungen, sondern zudem aus unterschiedlichen Objektraumpositionen abgebildet. In der schematischen Darstellung in Abbildung 9 wird verdeutlicht, dass der linke und der rechte Bereich des Tisches aus unterschiedlichen Perspektiven [ $H_1$ ] und [ $H_2$ ] abgebildet wurden. Dadurch entsteht innerhalb des Bildraums ein Bruch in der Darstellung des Tisches, der vom Maler durch ein Tuch verdeckt wird. Ferner identifiziert LORAN in [LORAN 2006] zwei weitere Hauptpunkte ([ $H_3$ ] und [ $H_4$ ]) in CÉZANNEs Gemälde. Der strukturelle Aufbau des Bildraums mit den vier Hauptpunkten wird in Abbildung 9

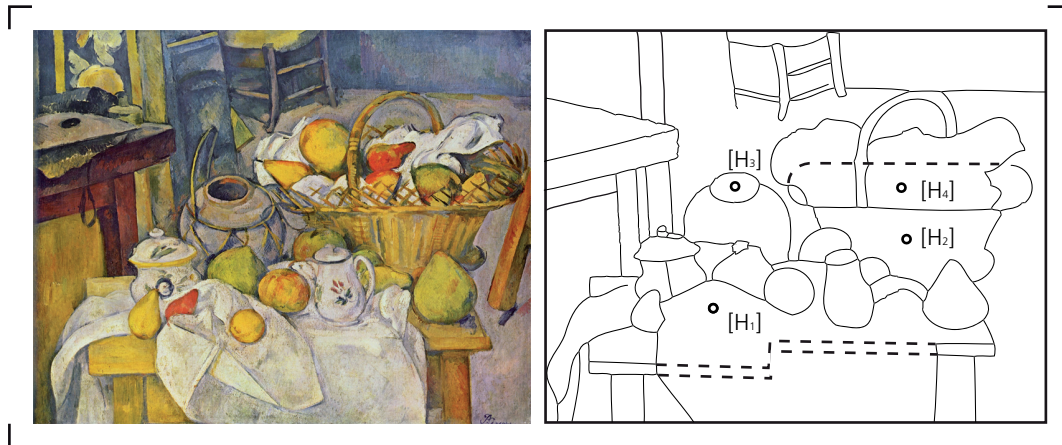


Abbildung 9: Objektraumbasierte Multiperspektive im Bild „Stilleben mit Obstkorb“, PAUL CÉZANNE, 1888-1890, aus [WENZEL 2005, S. 24] (links) und Skizze des Gemäldes nach [LORAN 2006] mit den Hauptpunkten des Bildes ([H<sub>1</sub>] bis [H<sub>4</sub>])

skizziert. Durch die zusätzlichen Perspektiven werden der Obstkorb und die Vase aus einer erhöhten Position, im Vergleich zur Augpunktposition für die Darstellung des Tisches, in den Bildraum überführt. In der Folge entsteht aus den erhöhten Positionen eine relative Draufsicht auf den Obstkorb sowie die Vase. Mit dieser Komposition von zentralprojektiven Bildfragmenten, in denen Standpunkt und Ausrichtung variieren, ist es CÉZANNE gelungen, Verdeckungen zu minimieren und eine exklusive Sicht auf wesentliche Bereiche des Objektraums zu erzeugen. GOMBRICH stellt in diesem Zusammenhang heraus, dass der Künstler zugunsten des optischen Eindrucks von Körperlichkeit und Raumtiefe auf die strenge Anwendung einer monoperspektivischen Darstellung verzichtet (vgl. [GOMBRICH 1996, S. 548]).

### Die bildraumbasierte Multiperspektive

Während bei der objektraumbasierten Multiperspektive eine Komposition in Abhängigkeit vom strukturellen Aufbau des Objektraums erfolgt, wird die bildraumbasierte Multiperspektive durch eine Komposition von Teilbildern auf Basis des Bildraums verwirklicht. Im Gegensatz zur objektraumbasierten Multiperspektive werden, ohne die explizite Einbeziehung des strukturellen Aufbaus des Objektraums, erzeugte Einzelbildräume in einem Bild vereint. Dabei kann diese Ausprägung multiperspektivischer Bilder zwischen zwei informalen Bildstrukturen unterschieden werden. Eine *kontinuierliche, multiperspektivische Darstellung* verbindet einzelne Perspektiven fließend, wodurch sich die Elemente im Bildraum zu einem geschlossenen Bildeindruck zusammenfügen, während eine *diskontinuierliche multiperspektivische Darstellung* eine zersplitterte Bildstruktur bewirkt.

Das im Jahr 1953 entstandene Bild „Relativität“ von MAURITS CORNELIS ESCHER (1898-1972) wirkt in deren Bildstruktur zusammenhängend, scheint aber einen unmöglichen Raum darzustellen (siehe Abbildung 10-links). Eine genauere Betrachtung zeigt, dass nicht die Geometrie der Objekte die visuelle Wahrnehmung stört, sondern die Semantik des Bildraums. Eine Nachbildung des dargestellten Raumes ist konstruierbar, wobei in diesem das Gesetz der Gravitation scheinbar keinen Einfluss hat. Das Bild setzt sich aus drei Teilbildern zusammen, sodass ein Gesamtbild mit dem Anschein entsteht, dass drei verschiedene Gravitationsfelder



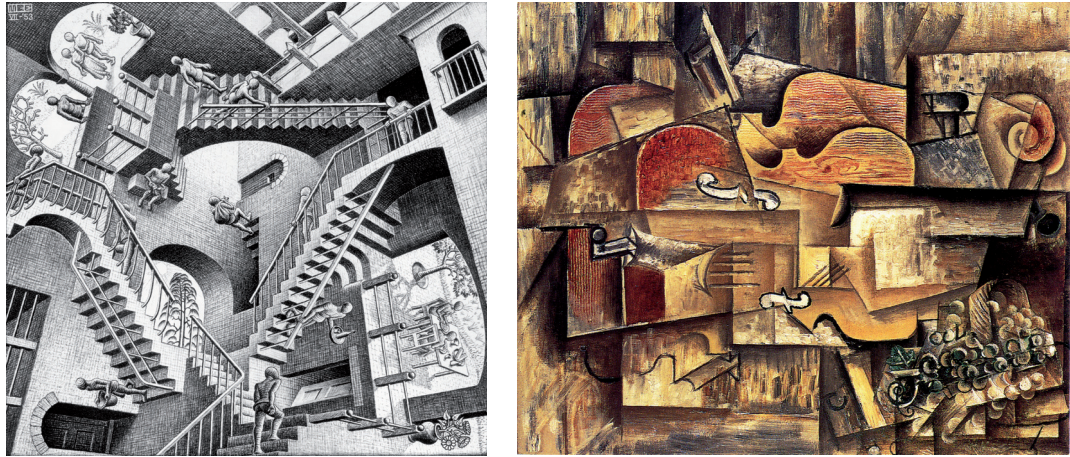


Abbildung 10: Bildraumbasierte Multiperspektive in Bildern. „Relativität“, MAURITS CORNELIS ESCHER, 1953, aus [ERNST & LOCHER 1983, S. 38] (links) und „Stilleben mit Geige und Trauben“, PABLO PICASSO, 1912, aus [GOMBRICH 1996, S. 574] (rechts)

im rechten Winkel zueinander auf die Personen im Bild wirken. Jedes Teilbild besitzt die bildstrukturellen Eigenschaften, wie diese durch eine zentralprojektive Abbildung erreicht werden. Die Komposition der drei Perspektiven, die untrennbar in einem Bild zusammengefügt sind, ermöglicht durch homogene Bildübergänge einen gleichzeitigen Einblick in drei unterschiedliche „Welten“. Die orthogonal zueinander ausgerichteten Szenerien ergeben sich durch die unterschiedlichen Augpunkte des Künstlers, wodurch ein Blick um Ecken und hinter Objekte erreicht wird, infolgedessen eine simultane Sicht in unterschiedliche Richtungen freigegeben ist.

Ein diskontinuierliches multiperspektivisches Bild zergliedert einzelne Ansichten in einer Form, dass die Geschlossenheit der Kontur von Objekten und deren Umraum aufgehoben wird. Es entsteht der Eindruck einer Dekonstruktion des Objektraums auf der Bildfläche. Zu Beginn des 20. Jahrhunderts fand in der Kunstrichtung des Kubismus diese Form multiperspektivischer Bilder eine Anwendung. Die abgebildeten Gegenstände wurden im Hinblick auf deren besondere Charakteristiken dargestellt, indem die essenziellen Bereiche auf der Bildfläche komponiert wurden (vgl. [BUDE U. A. 1988]). Wie das „Stilleben mit Geige und Trauben“ von PABLO PICASSO (1881-1973) in Abbildung 10 zeigt, erfolgte die Zergliederung in charakteristische Sichtweisen. Die dargestellten Objekte sind vorwiegend Gegenstände des Alltags; in PICASSOS Gemälde ein Musikinstrument und eine Fruchtschale. Ungewöhnliche Objekte wären vom Betrachter durch den teilweise hohen Deformationsgrad nicht erkannt worden (vgl. [GOMBRICH 1996, S. 574]). Der Prozess des Wiedererkennens der segmentierten Bildmotive ist jedoch notwendig, um einem Betrachter die Bildintention näher zu bringen (vgl. [BUDE U. A. 1988]). Aus angedeuteten, körperlosen Darstellungsfragmenten auf der Leinwand sollte sich nach GOMBRICH im geistigen Auge des Betrachters eine vollständige, greifbare und körperliche Vorstellung des Motivs zusammensetzen (vgl. [GOMBRICH 1996, S. 574 ff.]).

Die Nachahmung der Natur ist in dieser Darstellungsform nicht die primäre Bildintention. Vielmehr werden mithilfe dieser diskontinuierlichen bildraumbasierten Multiperspektive visuelle Botschaften geformt, die nicht unmittelbar aus dem Bild entnommen werden

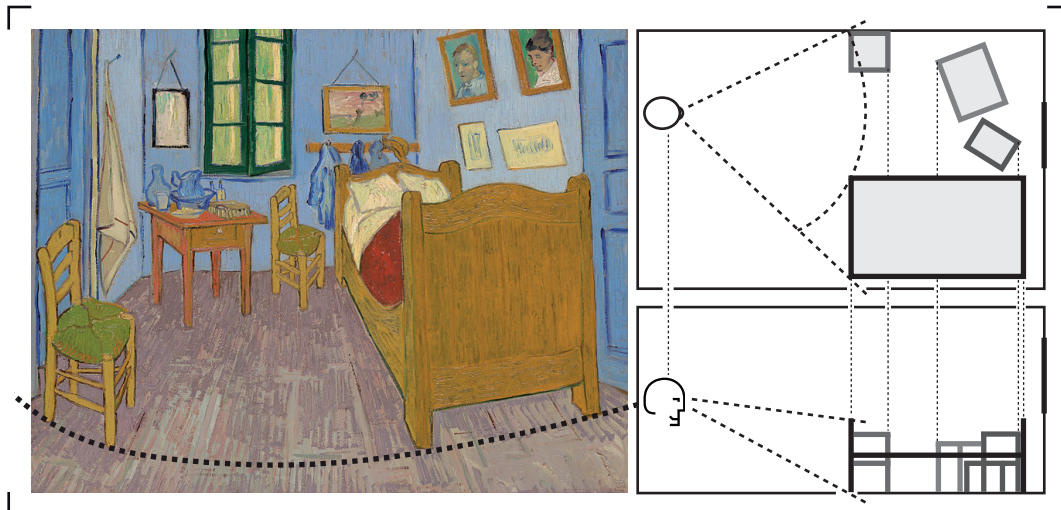


Abbildung 11: Krummlinige Darstellung im Bild „Schlafzimmer in Arles“, VINCENT VAN GOGH, 1888-1889, aus [GOMBRICH 1996, S. 549] erweitert durch eine extrahierte Kontur zur Verdeutlichung der krummlinigen Darstellungsweise (links) sowie die schematische Darstellung des Sichtfeldes in einer Draufsicht und einer Seitenansicht des Zimmers (rechts), nach [DURAND 2001].

können. Dazu ist es erforderlich, unterschiedliche Augpunktpositionen einzunehmen, um die in den Bildern dargestellten Motive zu interpretieren. Diese Pluralität der visuellen Positionen erfordert nach [JUTZ & SCHLEMMER 1989, S. 20] „einen summierenden Blick“ des Betrachters, um den Bildinhalt zu erschließen.

### 3.2.2.2 Krummlinige Darstellungen

Eine zweite Gruppe nichtlinearer Darstellungsformen umfasst Bilder, die eine Krummlinigkeit im strukturellen Aufbau des Bildraums aufweisen. Deren Bildstruktur ist durch konvexe oder konkave raumführende Geraden charakterisiert. Repräsentative Ausprägungen krummliniger Darstellungen sind *Panorama-* und *Fischaugenbilder*. Während Panoramadarstellungen durch eine kontinuierliche Veränderung des Blickwinkels entlang einer Achse definiert sind, werden Fischaugendarstellungen über eine Veränderung der Blickrichtung in Abhängigkeit von der Distanz zum Bildmittelpunkt beschrieben (vgl. [SALOMON 2006, S. 147 ff. UND 166 ff.]). Die Anwendung einer krummlinigen Darstellungsweise zeigt sich in der bildenden Kunst in verschiedenen Kontexten, wobei sich vor allem drei Darstellungsintentionen identifizieren lassen.

#### Imitation des menschlichen Sichtfeldes

Durch den Einsatz einer krummlinigen Darstellung wird eine Imitation des menschlichen Sichtfeldes mit dem Ziel erzeugt, die visuelle Wahrnehmung des Menschen nachzubilden. Dabei werden anstelle gerader Objektkanten gebogene Linien abgebildet, um die visuelle menschliche Raumwahrnehmung – die Konkavität des Auges – zu imitieren (vgl. [ERNST & LOCHER 1983, S. 52]). Die Abbildung 11 zeigt das Gemälde „Schlafzimmer von Arles“ von VINCENT VAN GOGH (1853-1890), das ein Zimmer mit einer rechteckigen Grundfläche verbildlicht. In der Analyse des Raumes zeigt DURAND in [DURAND 2001, S. 68], dass der Stuhl

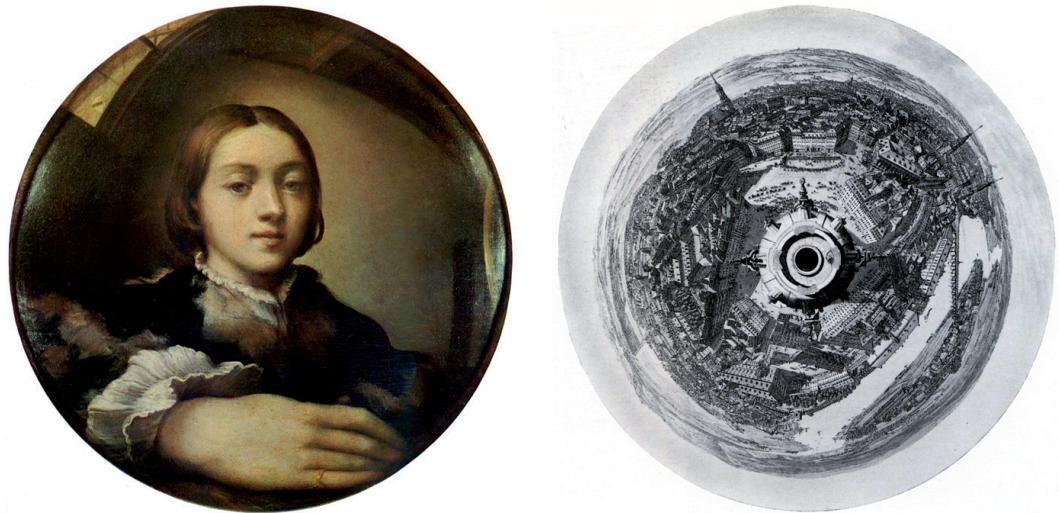


Abbildung 12: Krummlinige Darstellungsformen in Bildern. „Selbstporträt im konvexen Spiegel“, FRANCESCO MAZZOLA (PARMIGIANINO), 1524, aus [Geyer 1994, S. 101] (links) und „Panorama von Dresden“, CARL AUGUST RICHTER, 1824, aus [Hofmann 1998, S. 44] (rechts)

auf der linken Seite des Raumes orthogonal zum Bett verortet ist. Durch die extrahierte Kontur in Abbildung 11 wird deutlich, dass eine gekrümmte Geometrie für die Raumdarstellung auf der Fläche verwendet wird. Es ist davon auszugehen, dass der Künstler das Drehen des Kopfes beim Betrachten des Raumes im Bild festhielt und die Krummlinigkeit in der Nachahmung des menschlichen Blickes begründet ist.

### Lenkung der Aufmerksamkeit

Eine bewusst „unnatürliche“ Darstellungsweise unter Einsatz krummliniger Abbildungsmethoden war eine zweite Intention der Künstler (vgl. [FISCHER 1996, S. 92]). Für die Erzeugung der Werke wurden komplexe Abbildungstechniken angewendet, die mithilfe von Spiegelkugeln, Hohl- und Wölbspiegeln partielle Vergrößerungen erzeugten. Die hervorgerufenen Verzerrungen bildeten die Grundlage für die krummlinigen Bildstrukturen in den Kunstwerken. Das Gemälde „Selbstporträt im konvexen Spiegel“ von GIROLAMO FRANCESCO MARIA MAZZOLA (1503-1540, genannt PARMIGIANINO) in Abbildung 12 (links) zeigt das Spiegelbild des zeichnenden Malers. Durch den konvexen Spiegel wird die rechte Hand, die sich nah am Spiegel befindet, stark vergrößert dargestellt. Dadurch nimmt die Hand gegenüber dem Kopf respektive dem Gesicht eine hervorgehobene Stellung im Bild ein. Mit dieser künstlerischen Technik wurde versucht, durch die gezielte Abweichung von der zentralprojektiven Darstellung die Aufmerksamkeit des Betrachters zu lenken. Diese Darstellungsform soll nach HOFMANN den Betrachter „verblüffen, überraschen und verunsichern“ [HOFMANN 1998, S. 101].

### Erweiterung des Sichtfeldes

Die Erweiterung des Sichtfeldes ist ein dritter Anwendungskontext der krummlinigen Darstellungsweise. Die Abbildung 12 (rechts) zeigt das Gemälde „Panorama von Dresden“ von CARL AUGUST RICHTER (1785-1853). Der bildstrukturelle Aufbau des Bildes entspricht einer Fischaugendarstellung und zeigt den Dresdner Neumarkt in einem 360-Winkel-



grad-Panorama mit der Dresdner Frauenkirche als zentrales Bildelement. Der Künstler verwendet diese Darstellungsform, um dem Betrachter bei der Sicht auf den Dresdner Neumarkt eine möglichst große Menge an Informationen durch das Bild zu vermitteln. Das Gemälde stellt aufgrund der krummlinigen Perspektivdarstellung neben der Frauenkirche auch deren umliegende Häuserfronten sowie die nähere Umgebung um das zentrale Kirchengebäude dar. Während eine zentralprojektive Darstellung eine Draufsicht auf die Häuser in einem weitaus kleineren Sichtbereich abbilden würde, sind die Häuser in RICHTERS Gemälde in teilweiser Frontalansicht dargestellt. Dadurch wird eine Sicht auf den Dresdner Neumarkt erreicht, die dem Betrachter zusätzliche Einblicke in das Stadtbild bietet.

### 3.2.3 Lineare und nichtlineare Darstellungsformen in 3D-Anwendungen

Die Ausführungen zeigen eine Vielfalt von Darstellungsformen unabhängig von der Entstehungszeit und dem Erzeugungsmedium. Darüber hinaus wird deutlich, dass im Bereich der bildenden Künste der ausschließliche Gebrauch einer linearen Darstellungsweise kaum Anwendung findet. Dies mag durch die Entstehungszeit des jeweiligen Werkes begründet sein, da der zum Zeitpunkt der Fertigstellung eines Werkes vorherrschende Wissensstand oder die verwendeten Werkzeuge eine geometrische Korrektheit in der Bildstruktur nicht ermöglichten (vgl. [ALBERTI 1972; DÜRER 1525]). Dem entgegenstehend zeigt HOCKNEY in [HOCKNEY 2006], dass mit der Lochkamera ein Hilfsmittel den Künstlern zur Verfügung stand, um geometrisch korrekte lineare Abbilder zu erzeugen. Vielmehr zeigen die Betrachtungen, dass es sich um einen bewussten Bruch mit der reinen Zentralprojektion handelt. GOMBRICH veranschaulicht in [GOMBRICH 1960], dass die Aussagen, welche die Künstler vergangener Epochen durch die Bilder kommunizieren wollen, mannigfaltig waren: Ausdruck und Abstraktion, Symbolik und die Wahrnehmungswelt sind nur einige dieser Aspekte. Dabei dienten die unterschiedlichen Formen multiperspektivischer und krummliniger Darstellung zur Verbildlichung einer intendierten Aussage. Die Ausführungen zeigen ferner die Möglichkeiten und den zusätzlichen Nutzen auf, die Regeln und Methoden der Künstler und Maler für die Darstellung von Daten und Informationen in interaktive 3D-Anwendungen zu überführen. Diese Bestrebung wird durch den Umstand begünstigt, dass einerseits die Vertreter der neuzeitlichen Malerei zur Verwirklichung ihrer Werke die Lochkamera als Werkzeug einsetzen (vgl. [HOCKNEY 2006]), andererseits dieses Werkzeug als einfachste Form eines Abbildungsapparates zum Vorbild für die computergrafische Kamera wurde. Dies legt die Schlussfolgerung nahe, dass unabhängig von historischen und theoretischen Hintergründen der neuzeitlichen Malerei, die Methoden und Techniken für eine dialogorientierte Bildgestaltung in 3D-Benutzerschnittstellen interaktiver Systeme genutzt werden sollten.

## 3.3 Die computergrafische Bildsynthese

Eine Betrachtung der Verfahrensschritte innerhalb der Bildsynthese ist zweckmäßig, um den Projektionsschritt als einen Teilaspekt des Erstellungsprozesses computergrafischer Bilder nachzuvollziehen. Hierzu wird eine konzeptionelle Beschreibung der Rendering-Pipeline,

basierend auf den Ausführungen in [AKENINE-MÖLLER U. A. 2008, KAP. 2] zugrunde gelegt, da AKENINE-MÖLLER eine Referenzabfolge der Einzelschritte für eine Bilderzeugung aufzeigt. Hierbei soll ein generelles Verständnis für den Erstellungsprozess im Vordergrund stehen. Für eine detaillierte Erläuterung der einzelnen, konkreten Berechnungsvorschriften wird auf die entsprechende computergrafische Fachliteratur verwiesen.

### 3.3.1 Der Abbildungsprozess als Grafikpipeline

Das Modell der Grafikpipeline besteht aus drei Hauptverfahrensschritten mit den Termini *Anwendung*, *Geometrie* und *Rasterisierung*. Der Erstellungsprozess eines Abbildes wird in Abbildung 13 schematisch veranschaulicht und zeigt, dass die Abbilderzeugung ausgehend von einem Anwendungssystem erfolgt. Dieses System kann beispielsweise eine Applikation zur Interaktion mit einer virtuellen Umgebung oder zur Analyse und Auswertung wissenschaftlicher Daten sein (siehe Unterabschnitt 3.1.3). Durch die Vielfalt an existierenden Softwaresystemen lässt sich der erste Schritt der Bildsynthese nicht in einem sequenziellen Modell innerhalb der Grafikpipeline beschreiben. Generell werden in diesem Schritt die Primärgeometrie erstellt oder Änderungen an einer bestehenden visuellen Struktur vorgenommen. Zusätzlich werden innerhalb dieses ersten Verfahrensschrittes die Parameter für die virtuelle Kamera als ein Element des Objektraums definiert. Durch die Bestimmung von Position und Ausrichtung des virtuellen Augpunktes wird die Sicht auf die visuelle Struktur spezifiziert. Anschließend wird die Primärgeometrie in den Geometrieschritt überführt.

Innerhalb des Geometrieschrittes erfolgt zunächst die *Modell- und Kameratransformation* (siehe Abbildung 13). In diesem Verfahrensschritt werden die Szenenelemente zunächst in ein globales Koordinatensystem und damit für die gesamte Szene einheitliches System überführt, da die im Objektraum verorteten Szenenelemente innerhalb der Bildsynthese zunächst in einem eigenen lokalen Koordinatensystem beschrieben sind. In einem nachgelagerten Schritt wird die Szene in das Kamerakoordinatensystem überführt, wodurch die Szenendaten relativ zur bereits festgelegten computergrafischen Kamera definiert werden. Im anschließenden Schritt *Vertex-Shading und Beleuchtung* werden die *Vertices* der Szenenobjekte modifiziert. In der 3D-Computergrafik ist ein Vertex ein Eck- beziehungsweise Scheitelpunkt eines computergrafischen Körpers, der durch Positions- und Farbwertangabe sowie durch Transparenzwert, Normalenvektor und Texturkoordinaten definiert ist. Für eine Veränderung dieser Parameter werden sogenannte *Vertex-Schattierer* verwendet, die eine Modifikation an einzelnen Vertices umsetzen und dadurch den Einfluss von Beleuchtung und Texturierung auf das Element verändern. Im nachfolgenden *Projektionsschritt* wird die Abbildung der dreidimensionalen Szene auf eine zweidimensionale Fläche vollzogen. Die Abbildungsvorschrift entspricht in computergrafischen Anwendungen in der Regel der Parallel- oder der Zentralprojektion, welche mithilfe des computergrafischen Kameramodells spezifiziert ist. Darüber hinaus existieren weitere Projektionsverfahren, die auf Basis eines modifizierten computergrafischen Kameramodells ausgeführt werden können (siehe Unterabschnitt 3.3.2). Um eine ressourcenschonende Bildsynthese sicherzustellen, ist es zweckdienlich, die Anzahl der darzustellenden Objekte zu minimieren. Dies wird im Verfahrensschritt des *Clippings* erreicht und erfolgt auf Basis des *Kamerasichtkörpers*. Die Objekte, die

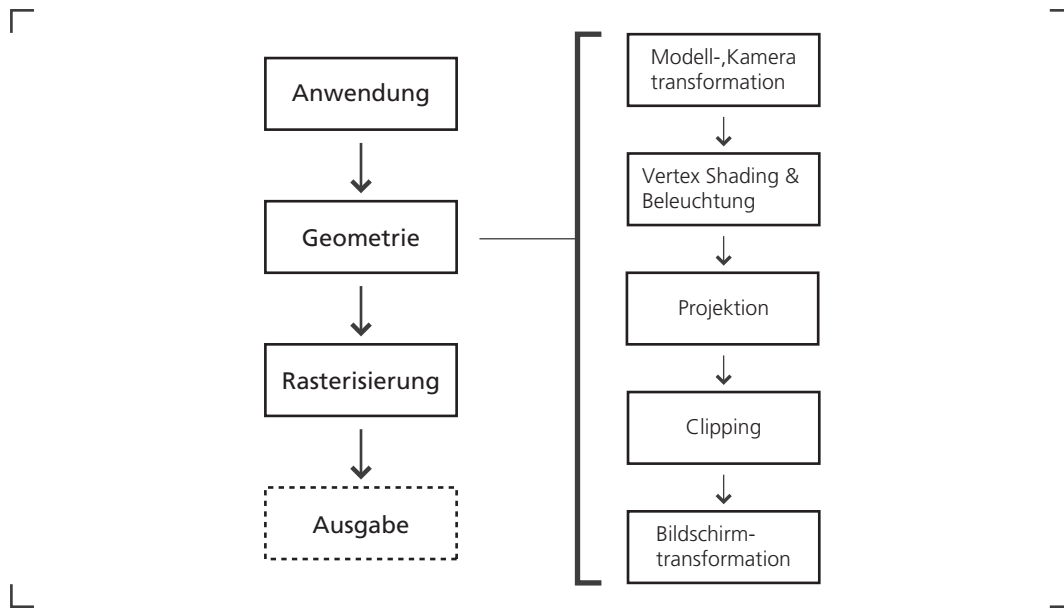


Abbildung 13: Modell der Grafikpipeline für die computergrafische Bildsynthese (nach [AKENINE-MÖLLER U. A. 2008, KAP. 2])

sich außerhalb des definierten Sichtvolumens befinden, werden für das weitere Rendering verworfen, wodurch der Berechnungsaufwand zur Erzeugung des Bildraums reduziert wird. Im letzten Schritt innerhalb des Geometrieschrittes wird die *Bildschirm-Transformation* ausgeführt, um das Abbild der dreidimensionalen Szene auf einen Zielbereich des Ausgabegerätes festzulegen.

Der dritte Schritt innerhalb der Bildsynthese ist der *Rasterisierungsschritt* (siehe Abbildung 13). Dieser umfasst, für eine Darstellung der visuellen Struktur auf dem Ausgabegerät, die Überführung der geometrischen Objekte in diskrete Pixelbilder. Dazu erfolgt eine Umwandlung der Objekt-Vertices in eine Pixelgrafik. Dabei werden alle grafischen Primitive gerastert und die zugehörigen Pixel auf dem Ausgabegerät entsprechenden eingefärbt. Die Farbe eines Pixels wird durch die Beleuchtung, die Textur und die weiteren Materialeigenschaften der Vertices bestimmt. Innerhalb dieses Verfahrensschrittes kann eine Veränderung der Farbe, der Textur, der Beleuchtung und der Sichtbarkeit durch Softwarebausteine, die *Fragment-Schattierer*, erreicht werden. Diese Schattierer bieten durch die Anwendung von Bildfiltern die Möglichkeit pixelbasierte Manipulationen am Abbild vorzunehmen, um beispielsweise Tiefenunschärfe oder Farbfilter im Bildraum zu realisieren. Mit Abschluss des Rasterisierungsschrittes wird das Abbild auf dem Ausgabegerät dargestellt.

Die Verfahrensschritte der Grafikpipeline sind mit dem Fokus auf eine naturalistische beziehungsweise fotorealistische Darstellung konzipiert, realisiert und unter dem Aspekt der Berechnungsgeschwindigkeit optimiert. Die Arbeiten aus dem Bereich des Non-Photorealistic Rendering verdeutlichen aber, dass die computergrafische Rendering-Pipeline die Erzeugung nicht-fotorealistischer Abbilder unterstützt. Durch eine Anpassung der Bildsynthese werden Darstellungen erzeugt, die beispielsweise Strichzeichnungen (vgl. [DEUSSEN 2001, S. 74 FF.]), Aquarellen (vgl. [COCONU U. A. 2006, S. 28 FF.]) und Ölgemälden (vgl. [AGRAWALA U. A. 2000, S. 13]) nachempfunden sind und folglich keiner fotorealistischen Darstellung entsprechen. In

weiteren Arbeiten wird darüber hinaus der praktische Nutzen von nicht-fotogleichen Visualisierungen unter Verwendung nicht-fotorealistischer Beleuchtung zur besseren Darstellung von Objektformen (vgl. [GOOCH U. A. 1998]), durch Texturierung für eine Verstärkung der räumlichen Tiefenwirkung (vgl. [LUFT U. A. 2006]) und durch Farbgebung für die Identifikation und Unterscheidung von Objekten (vgl. [MITCHELL U. A. 2007]) aufgezeigt.

Die verfahrenstechnischen Veränderungen des Bildsyntheseprozesses, die zu einer Veränderung der bildstrukturellen Eigenschaften eines computergrafisch erzeugten Bildes führen, werden bisher überwiegend unter den Aspekten der Machbarkeitsprüfung (vgl. [BROSZ U. A. 2007; SINGH 2002]) und der Reduzierung zentralprojektiver Darstellungsdefizite betrachtet (vgl. [ZELNIK-MANOR U. A. 2005; CARROLL U. A. 2010]). Eine Veränderung der bildstrukturellen Eigenschaften kann im Geometrieschritt durch den Einsatz von Vertex-Schattierern erfolgen und im Rasterisierungsschritt mittels Fragment-Shader vollzogen werden. Beispielsweise zeigen SPINDLER U. A. in [SPINDLER U. A. 2006], dass durch den Einsatz von Schattierern Fokus- und Kontext-Darstellungen, basierend auf den Geometriedaten des Objektraums und des Bildraums, realisiert werden können. Darüber hinaus zeigen Arbeiten, die einen bildbasierten Modifikationsansatz wählen, die Verwirklichung von Panorama- (vgl. [YANG U. A. 2005]) und Fischaugendarstellungen (vgl. [KEAHEY 1998]), aber auch von verzerrungsfreien Darstellungen [ZORIN & BARR 1995; CARROLL U. A. 2009].

Ein großer Teil wissenschaftlicher Arbeiten, in denen Herausforderungen in Bezug auf den bildstrukturellen Aufbau von Bildräumen adressiert werden, nutzt jedoch den Projektionsschritt zur Modifikation des Bildraums. Dies liegt darin begründet, dass bildbasierte Verfahren lediglich über den zweidimensionalen Bildraum als Ausgangsbasis verfügen. Durch die Rasterisierung steht für die bildstrukturelle Veränderung einer Visualisierung in der Folge nur eine begrenzte Anzahl an Bildpunkten zur Verfügung. Das kann bei einer Modifikation der Eigenschaften zu Darstellungsfehlern oder fehlenden Bildpunkten führen. Mit einer vergrößerten Anzahl von Bildpunkten können diese Fehler gemindert werden, wobei sich der Berechnungsaufwand infolgedessen erhöht. Eine weitere Einschränkung entsteht aus der Reduktion der Informationsmenge, die durch den Abbildungsschritt einer dreidimensionalen Szene auf eine zweidimensionale Fläche hervorgerufen wird. Beispielsweise sind bildbasierte Verfahren zur Modifikation der Sichtbarkeit von Objekten aufgrund der fehlenden Tiefeninformationen nur eingeschränkt anwendbar. Ferner ist die Identifikation expliziter Objekte und spezieller Szenenparameter ausgehend vom Bildraum limitiert. Infolgedessen wird für die Analyse des Gestaltungspotenzials der computergrafische Projektionsschritt betrachtet. SINGH weist in diesem Zusammenhang darauf hin, dass der Fokus einer Veränderung im Abbildungsprozess hierbei auf der Gestaltung interaktiver 3D-Anwendungen liegen sollte: *„Past experiments with non-linear perspectives have primarily focused on abstract mathematical camera models for raytracing, which are both noninteractive and provide the artist with little control over seeing what he wants to see.“*<sup>45</sup> [SINGH 2002, S. 17]. Durch eine Untersuchung der Modifikationsmöglichkeiten, die zur Erzeugung von linearen und nichtlinearen Darstellungsweisen genutzt werden, können Methoden und Techniken für den Entwurf von

<sup>45</sup> „Frühere Untersuchungen nicht-linearer Perspektiven haben sich überwiegend auf abstrakte mathematische Modelle für das Raytracing fokussiert, die nicht interaktiv waren und den Künstler nur wenig Kontrolle über die Ausgestaltung gaben.“

3D-Benutzerschnittstellen identifiziert werden. Brosz bekräftigt die Notwendigkeit einer Bereitstellung von verschiedenen Darstellungsmethoden in interaktiven Anwendungen wie folgt: „*In addition to the standard [...] projections, there have been a number of alternate projection models proposed. However, [...] the standard projections remain commonly used because they fit within a well understood geometric framework that is readily implementable.*“<sup>46</sup> [BROSZ U. A. 2007, S. 33]. Daher werden zunächst das computergrafische Kameramodell und nachfolgend dessen Modifikationsmöglichkeiten analysiert.

### 3.3.2 Das computergrafische Kameramodell

Zur Beschreibung des Projektionsschrittes wird das computergrafische Kameramodell eingesetzt, wie es in Abbildung 14 schematisch dargestellt ist. Ausgehend vom Verfahren der Zentralprojektion (siehe Unterabschnitt 3.2.1) abstrahiert die Kamera ein unbewegtes Auge, das in einen virtuellen Objektraum sieht. Das *Zentrum der Projektion* bildet den *Augpunkt* der Kamera [A], während die *optische Achse* beziehungsweise der *Sichtvektor* [S] die Blickrichtung festlegt. Darüber hinaus wird durch den *Aufwärtsvektor* [U] die Orientierung der Kamera und durch den *Sichtkörper* der darzustellende Ausschnitt des Objektraums definiert. Der Kamerasichtkörper wird durch die *vordere* [N] und die *hintere Begrenzungsebene* [F] festgelegt, wodurch der Projektionsvorgang auf Objekte innerhalb des Sichtkörpers beschränkt wird. Im Unterschied zum realen Vorbild – der Lochkamera – ist im computergrafischen Kameramodell die Bildebene zwischen dem Zentrum der Projektion und der Szenerie verortet. Durch die Verbindung der Objektpunkte einer Primärgeometrie mit dem Zentrum der Projektion entsteht auf der Bildebene das Abbild. Das in Abbildung 14 dargestellte Abbild, als das Resultat des computergrafischen Projektionsschrittes, bietet in 3D-Anwendungen überwiegend eine zentralperspektivische Sicht entlang der optischen Achse mit dem Hauptpunkt des Bildes in der Abbildmitte.

Der Projektionsvorgang, der durch das Kameramodell beschrieben wird, ist hinsichtlich einer effizienten Berechnung und Darstellung des Abbildes auf dem Ausgabegerät optimiert. Aufgrund dessen ist die Kamera über jene Parameter spezifizierbar, die für eine Berechnung des Abbildungsvorganges zwingend erforderlich sind. Trotz der Beschränkung durch eine minimierte Anzahl an Spezifikationsparametern existieren Methoden, den Abbildungsprozess zu beeinflussen. Um die Darstellungsmöglichkeiten aufzuzeigen, erfolgen, aufbauend auf den Erläuterungen zur *Camera obscura* und deren Funktionsweise in [FRANKE U. A. 2005, S. 4], eine Betrachtung modifizierter Kameramodelle und die aus dem Projektionsvorgang resultierenden Bildräume. Dazu werden die Modifikationsmöglichkeiten des computergrafischen Kameramodells (siehe Abbildung 14) im Hinblick auf das Potenzial zur Gestaltung computergrafischer Bildräume betrachtet:

- Die Transformation der *Objektgeometrie* [O] in Abhängigkeit von den Kameraparametern führt gegenüber einer konventionellen zentralprojektiven Darstellung zu einem veränderten Abbild auf der Bildebene, da durch die Objektverformung strukturelle Änderungen im Bildraum entstehen (siehe 3.3.2.1).

<sup>46</sup> „Zusätzlich zu den standardmäßigen Projektionsverfahren existieren eine Reihe weiterer Abbildungsmethoden. Dennoch bleibt die Standardprojektion das vorwiegend eingesetzte Verfahren, da dieses auf einer verständlichen Methodik basiert, die leicht umsetzbar ist.“

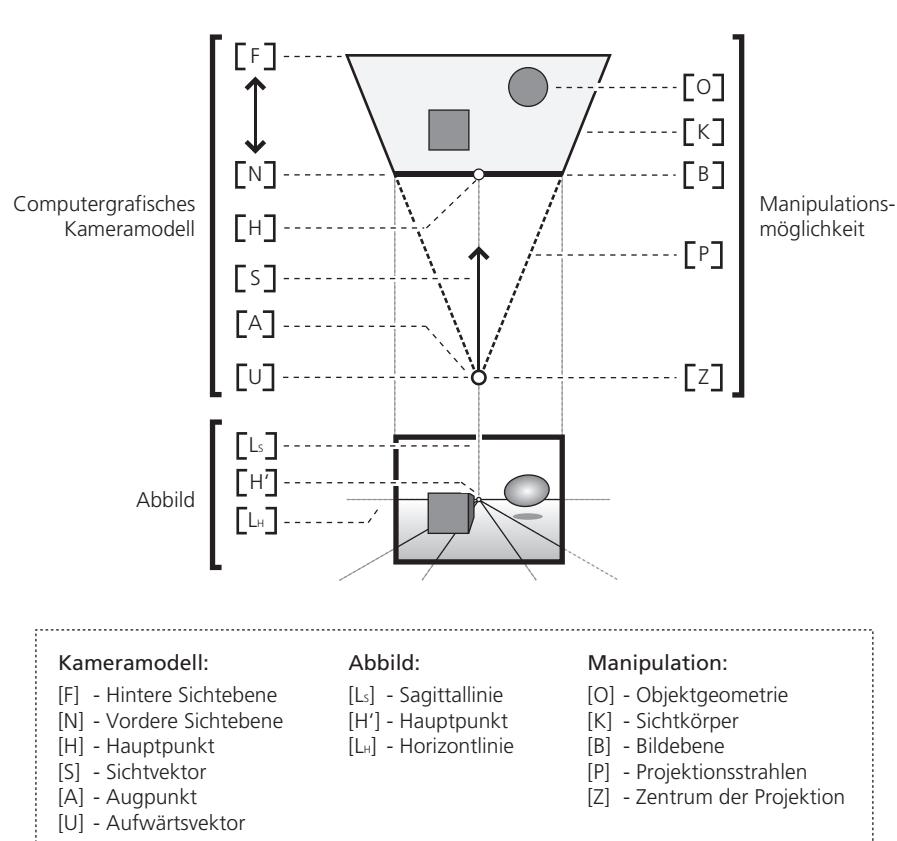


Abbildung 14: Kameramodell zur Erzeugung computergrafischer Abbilder bestehend aus dem Augpunkt [A], Sichtvektor [S], Aufwärtsvektor [U] sowie den durch vordere [N] und hintere Sichtbegrenzungsebene [F] definierten Sichtkörper und das durch die Projektion entstehende Abbild Hauptpunkt [H'] im Schnittpunkt von Horizont [L<sub>h</sub>] und Sagittallinie [L<sub>s</sub>]. Ferner sind die Möglichkeiten der Kameramodifikation aufgezeigt (Objektgeometrie [O], Sichtkörper [K], Bildebene [B], Projektionsstrahlen [P] und Zentrum der Projektion [Z]).

- Der Eingriff in die bildstrukturellen Eigenschaften eines Abbildes durch die Auflösung der Linearität der *Projektionsstrahlen* [P], bewirkt die Erzeugung krummliniger Darstellungen (siehe 3.3.2.2).
- Die *Bildebene* [B] kann zur Beeinflussung des Abbildungsprozesses dienen. Das Zusammenführen mehrerer computergrafischer Projektionsresultate auf einer Bildebene kann zur Erzeugung kompositorischer Bildräume genutzt werden (siehe 3.3.2.3).
- Eine weitere Modifikationsmöglichkeit bedient sich dem *Zentrum der Projektion* [Z]. Unter Verwendung mehrerer Zentren können durch den Abbilderzeugungsprozess *multiple Sichten* auf eine visuelle Struktur verwirklicht und zu einer gemeinsamen Darstellung montiert werden (siehe 3.3.2.4).
- Die fünfte Möglichkeit zur Veränderung des Projektionsschrittes ist die geometrische Abwandlung des *Kamerasichtkörpers* [K]. Durch eine Loslösung von der bestehenden Symmetrie des Kamerasichtkörpers werden die Positionierung des Hauptpunktes und die Perspektiveigenschaften des Abbildes variant und stehen somit als Mittel der Gestaltung zur Verfügung (siehe 3.3.2.5).

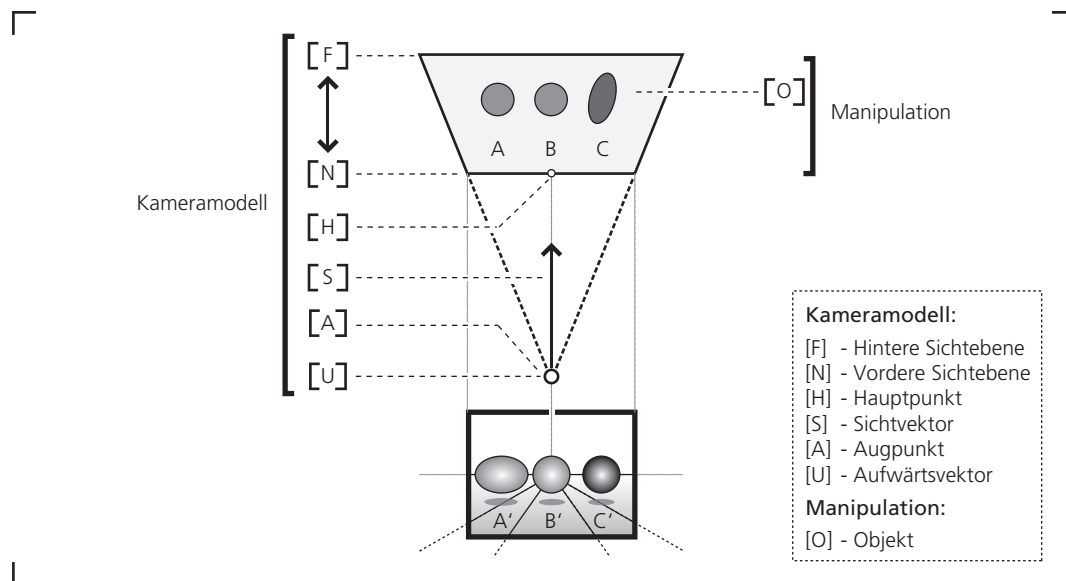


Abbildung 15: Kameramodell zur Erzeugung einer objektraumbasierten Multiperspektive auf Basis von Geometrieänderungen am Objekt [O] sowie das durch die Projektion entstehende Abbild des Objektraums.

### 3.3.2.1 Manipulation am Objekt

Ausgehend vom Anwendungsschritt wird die Primärgeometrie in der Bildsynthese aus dem Objektraum in den Bildraum überführt. Dabei unterliegen die visuellen Strukturen bei der zentralprojektiven Abbildung einer Gleichbehandlung ausgehend vom Projektionszentrum hin zur Bildperipherie. Die Ursache liegt in den Eigenschaften des monoperspektivischen Abbildungsprozesses begründet, wodurch die Geometrie des erzeugten Bildraums einem singulären Projektionszentrum unterworfen ist. Das Aufbrechen dieser einheitlichen Darstellungsweise eröffnet Gestaltungsmöglichkeiten durch eine differenzierte Abbildung der Szenenelemente im Bildraum. Für eine Akzentuierung einzelner Szenenelemente „mit dem Ziel mehrere Bereiche von dialoghafter Bedeutsamkeit in einem Bild zu etablieren, mehr noch, narrativ zu vereinigen“ ([FRANKE U. A. 2005, S. 498]), erfolgt die Veränderung der Objektgeometrie in Abhängigkeit von der Kamera. Eine visuelle Betonung von Elementen und Bereichen im Bildraum wird beispielsweise in [SINGH & BALAKRISHNAN 2004] angewendet, indem fokussierte Szenenbereiche in Abhängigkeit von der Kamera skaliert werden. Durch eine vergrößerte Darstellung der fokussierten Objekte wird die Relevanz der gewählten Objekte implizit hervorgehoben. RADEMACHER nutzt in [RADEMACHER 1999] einen vergleichbaren methodischen Ansatz, um die Charakteristiken von Objekten in Abhängigkeit zur Kamerabewegung herauszustellen, indem die Elemente sichtabhängig verformt werden. Eine weitere Technik wird in [FRANKE U. A. 2007] vorgestellt, bei der eine Geometrieänderung zur Erzeugung einer objektraumbasierten Multiperspektive realisiert wird. Bei dem Verfahren von FRANKE U. A. wird der Abstand eines Objektes von der optischen Achse der Kamera bestimmt, um aufbauend darauf eine geometrische Transformation des Objektes zu vollziehen. Dieser Vorgang führt zu einer Darstellung des Objektes, welche die bildstrukturellen Charakteristiken einer separaten, zusätzlichen Perspektive im Bildraum aufweist.



Die Abbildung 15 zeigt das Kameramodell zur Erzeugung einer kameraabhängigen Geometriemanipulation in Anlehnung an die Methode von FRANKE. Die geometrische Transformation des Objektes [C] in Abbildung 15 verdeutlicht, dass die Veränderung der Objektgeometrie im Objektraum zu einer gestaltungswahrenden Objektgeometrie im Bildraum führt. Auf Basis dieser Veränderung des Abbildungsprozesses können ohne eine permanente geometrische Abwandlung der Elemente des Objektraumes dialogisch bedeutsame Objekte (vgl. [GROH 2007, S. 34]) akzentuiert werden. Gleichzeitig ergibt sich die Möglichkeit projektionsbedingte Verzerrungen der Objekte zu unterbinden (vgl. [FRANKE U. A. 2007, S. 119–120]). Folglich können durch die kameraabhängige Objektmanipulation Abbilder erzeugt werden, deren bildstrukturelle Eigenschaften eine Vergleichbarkeit von Szenenobjekten erreicht, wie diese im Gemälde MASACCIOs (siehe 3.2.2.1) identifiziert werden. Im Gegensatz zu einem statischen Bild bedingt, im Kontext interaktiver Anwendungen, eine kameraabhängige Objekttransformation jedoch eine fortwährende Anpassung der visuellen Strukturen. Da durch Handlungen des Nutzers eine kontinuierliche, nutzerintendierte Neugestaltung des Bildraums in 3D-Anwendungen erfolgt, ist eine permanente Geometrieveränderung notwendig, um die relative Konstanz der Objekte im dynamischen Bildraum zu gewährleisten. Die Integration einer solchen kameraabhängigen Geometriemanipulation in eine interaktive Anwendung erfolgt in [WOJZIAK U. A. 2011A, S. 4–5] in Form einer objektraumbasierten Multiperspektive und wird weiterführend in [ZAVESKY U. A. 2010, S. 240–243], im Kontext der Produktvisualisierung, als Mittel der Gestaltung verdeutlicht.

### 3.3.2.2 Manipulation an den Projektionsstrahlen

Eine Veränderung des computergrafischen Abbildungsprozesses kann durch eine Manipulation an den Projektionsstrahlen erfolgen. Durch eine Krümmung wird deren, im konventionellen Abbildungsprozess inhärente Geradlinigkeit aufgehoben, wodurch eine krummlinige Darstellung auf der Bildebene erzeugt wird. Im Unterschied zu einer konventionellen computergrafischen Abbildung zeigt eine krummlinige Perspektive ein verändertes Größenverhältnis zwischen den visuellen Strukturen im Vorder-, Mittel- und Hintergrund des Bildraums. Während in zentralprojektiven Abbildern die inhärente Räumlichkeit des Objektraums in den Bildraum überführt wird, ist in krummlinigen Abbildern dieser „geöffnet“. Dadurch können die visuellen Strukturen in Weitwinkel- beziehungsweise Überblicksdarstellungen abgebildet werden. Die Visualisierungen weisen dabei bildstrukturelle Eigenschaften auf, die ebenso in der bildenden Kunst für Panoramadarstellungen und bei der Imitation des menschlichen Sichtfeldes eingesetzt werden (siehe 3.2.2.2). In [LÖFFELMANN & GRÖLLER 1996] wird gezeigt, dass ebenso in computergrafischen Bildern eine *kurvilineare Projektion* für eine Erweiterung des Sichtbereiches eingesetzt wird. Ferner zeigen die Arbeiten [WANG U. A. 2005] und [VALLANCE & CALDER 2001A], dass für Darstellungen zur Lenkung der Aufmerksamkeit (siehe 3.2.2.2) Fischaugenprojektionen verwendet werden, um ein Ordnen von Szenenelementen in Bezug auf deren Relevanz zu realisieren.

Die Verwirklichung der computergrafischen Darstellung kann dabei durch zwei unterschiedliche Verfahrensweisen vollzogen werden. Einerseits werden krummlinige Darstellungen durch eine Projektion auf nichtplanare Flächen erzeugt. Dies erfolgt, indem der Objektraum auf

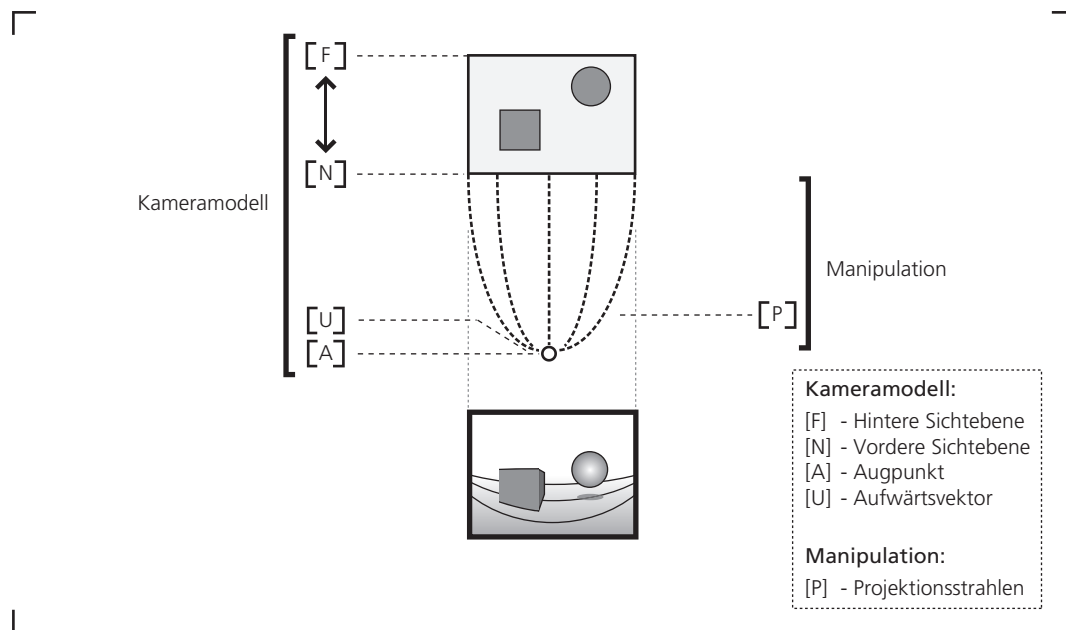


Abbildung 16: Kameramodell zur Erzeugung einer krummlinigen Perspektive durch kurvilineare Projektionsstrahlen [P] und das durch die Projektion entstehende Abbild des Objektraums.

einen Körper projiziert und in einem nachfolgenden Schritt mithilfe einer Abbildungsfunktion „aufgefaltet“ auf der Bildebene visualisiert wird. Andererseits erfolgen kurvilineare Projektionen unter Anwendung des Raytracing-Verfahrens (vgl. [GLASSNER 1989, S. 2]). Dieses Verfahren dient in der Computergrafik der Ermittlung der Sichtbarkeit von Objekten in computergrafischen 3D-Szenen. Ausgehend von einem Punkt im Raum erfolgt durch die Aussendung von Strahlen und mithilfe einer Methode der rekursiven Strahlverfolgung eine Sichtbarkeitsberechnung. Diese wird überwiegend für die globale Beleuchtung virtueller Szenen eingesetzt, kann aber darüber hinaus zur Erzeugung von krummlinigen Abbildern verwendet werden. Der Abbildungsvorgang wird hierbei durch eine Menge von Strahlen beschrieben, bei der jeder Projektionsstrahl einen Startpunkt und eine Richtung besitzt. Die Strahlen müssen für eine krummlinige Darstellung nicht zwingend die gleiche Ausgangsposition besitzen. Geometrische Figuren wie beispielsweise Tori oder Kugeln dienen in [LÖFFELMANN & GRÖLLER 1996, S. 220 ff.] als Emitter-Flächen zur Aussendung der Strahlen. Jedes Pixel auf der Bildebene wird einem Punkt auf den Elementen des Objektraums zugeordnet. Der vom jeweiligen Pixel ausgesendete Strahl erhält über die Geometriedaten der Emitter-Figur die Parameter für den Ursprung und die Ausrichtung. Das Zentrum der Projektion definiert sich bei LÖFFELMANN und GRÖLLER somit nicht als Punkt, sondern als geometrisches Objekt, von dessen Oberfläche die Projektionsstrahlen ausgesendet werden (vgl. [LÖFFELMANN & GRÖLLER 1996, S. 220 ff.]).

Computergrafische Panoramadarstellungen und Fischeugenansichten sind die generalisierten Ausprägungen der Bilder, die durch eine Manipulation der Projektionsstrahlen erzeugt werden [SALOMON 2006, KAP. 4]. Die Abbildung 16 zeigt in einer schematischen Form das Kameramodell zur Erzeugung krummliniger Abbilder. Dabei erfolgt die Projektion nicht auf Basis linearer Projektionsstrahlen, sondern die Strahlen werden durch eine spezifische

Abbildungsfunktion beschrieben. Diese legt die Form der Projektionsstrahlen fest, wodurch eine krummlinige Darstellung auf der Bildebene der computergrafischen Kamera erzeugt wird. Bildstrukturell unterscheiden sich die Darstellungen gegenüber einem zentralprojektiven Abbild durch die Form der in die Tiefe des Objektraums verlaufenden Objektkanten beziehungsweise Tiefenlinien. Weiterhin weisen Panorama- und Fischaugendarstellungen jeweils spezifische bildstrukturelle Eigenschaften auf. Während der Bildraum einer zylindrischen Panoramaprojektion (vgl. [SALOMON 2006, S. 167 ff.]) zwei Hauptpunkte aufweist, besitzt das Abbild einer Fischaugenprojektion (vgl. [SALOMON 2006, S. 147 ff.]) eine geometrische Mitte. Weiterhin werden Objektgeraden, die im Objektraum orthogonal zur Bildebene verlaufen in Fischaugendarstellungen geradlinig dargestellt, während in Panoramadarstellungen diese einen krummlinigen Charakter aufweisen. Beiden Darstellungsformen ist jedoch gemein, dass die Abbildungsfunktionen eine Überführung des gesamten Objektraums in den Bildraum ermöglichen, während durch eine zentralprojektive Darstellung lediglich ein Ausschnitt in den Bildraum projiziert werden kann. Die Möglichkeit einer umfassenden Sicht auf eine visuelle Struktur wird in der Arbeit von BAYARRI verdeutlicht, indem kurvilineare Projektionen konzipiert und in einer computergrafischen Anwendung implementiert werden (vgl. [BAYARRI 1995, S. 433 ff.]). Weiterhin wird die Echtzeitfähigkeit kurvilinearere Projektionen aufgezeigt und folglich der Einsatz krummliniger Darstellungen in interaktiven Anwendungen dokumentiert.

### 3.3.2.3 Manipulation an der Bildebene

Die computergrafische Bildebene ist die „Fläche“, auf die ein dreidimensionaler Objektraum projiziert wird. In der Computergrafik entspricht die vordere Begrenzung des Kamerasichtkörpers der Bildebene. Infolgedessen werden bei einer zentralprojektiven Projektion die innerhalb des Sichtkörpers befindlichen Szenenelemente auf der Bildebene abgebildet. Durch eine Bildebenenmodifikation kann jedoch eine Zusammenführung mehrerer Abbilder auf einer Bildebene erfolgen. Durch ein Arrangieren von Einzelabbildern wird der Bildraum komponiert, wodurch ein multiperspektivisches Abbild erzeugt wird. Dadurch kann, im Gegensatz zu einem zentralprojektiven Abbild, die Sicht auf Objekte und Szenenbereiche individuell festgelegt und für eine kontextbezogene Darstellung zusammengeführt werden. Diese Form der Bildkomposition wird in der Computergrafik als *Image Mosaicing*<sup>47</sup> bezeichnet und dient als Mittel der Gestaltung, um beispielsweise semantische Zusammenhänge zwischen räumlichen Objekten zu schaffen, ohne an den strukturellen Aufbau des Objektraums gebunden zu sein. Eine Bildebenenkomposition kann hierbei entweder mit Abbildern erfolgen, denen ein gemeinsamer Objektraum zugrunde liegt oder auf Basis einer kamerabezogenen Segmentierung der Primärgeometrie verwirklicht werden.

Bei einer Bildkomposition, bei der jedes Abbild auf Basis einer Teilmenge der visuellen Struktur erzeugt und auf der Bildebene verortet wird, weist das Gesamtbild die strukturellen Eigenschaften einer objektraumbasierten Multiperspektive auf. Das Gestaltungspotenzial zeichnet dadurch aus, dass die Szenenelemente durch individuelle Abbildungsprozesse dargestellt werden können (vgl. [WOJDAK U. A. 2011E, S. 6–7]). AGRAWALA verwendet in

<sup>47</sup> *Image Mosaicing* engl. Vorgang der Mosaikbilderstellung; Der Vorgang ermöglicht die Erzeugung eines Bildes durch die Zusammenführung vieler Einzelbilder

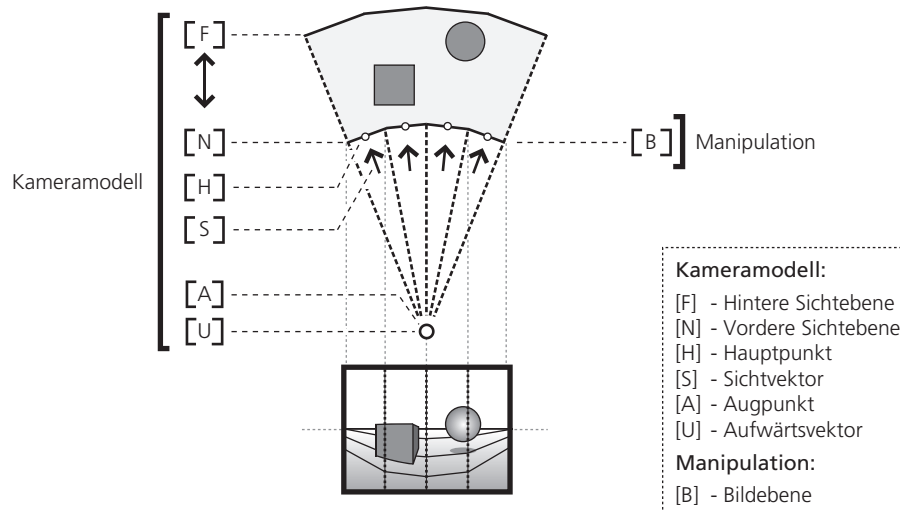


Abbildung 17: Kameramodell zur Erzeugung einer bildraumbasierten Multiperspektive durch die Komposition von Einzelbildebenen [B] nach dem Prinzip der Streifenkamera und das durch die Projektion entstehende Abbild des Objektraums

[AGRAWALA U. A. 2000, S. 229 ff.] diesen Ansatz um Objekte aus unterschiedlichen Perspektiven abzubilden und mit der jeweils charakteristischen Seite in einem Bild anzuordnen. Es zeigt sich dadurch, dass aufgrund der individuellen Darstellung von Elementen deren Gestalt und die Struktur von 3D-Szenen bei gleichzeitig geringerer Objektverdeckung erreicht werden können [AGRAWALA U. A. 2000, S. 134–135]. Im Unterschied zur objektabhängigen Bildebenenmanipulation kann eine Komposition ferner durch eine Zusammenführung von Teilbildern auf Basis einer gemeinsamen Primärgeometrie erfolgen. Diese Ausprägung der Bildebenenmanipulation ermöglicht eine bildraumbasierte multiperspektivische Darstellung und kann dazu dienen, sukzessive Zusammenhänge zwischen Inhaltsbereichen simultan darzustellen. Die Bilder, die durch diesen Modifikationsansatz erzeugt werden können, besitzen bildstrukturelle Eigenschaften wie diese anhand der Werke von ESCHER und PICASSO in Abschnitt 3.2.2.1 veranschaulicht werden. Eine Verdeutlichung der Funktionsweise auf Basis des computergrafischen Kameramodells bietet das Prinzip der Streifenkamera als Umsetzung des Mosaikverfahrens (siehe Abbildung 17). Eine Streifenkamera besteht aus mehreren Einzelkameras, deren Abbilder in der Zusammenführung ein Gesamtbild erzeugen, das einen simultanen Blick in unterschiedliche Richtungen freigibt oder ein Objekt gleichzeitig von verschiedenen Seiten zeigt. RADEMACHER und BISHOP verwenden diesen Ansatz in [RADEMACHER & BISHOP 1998], um alle Seiten eines Objektes simultan in einem Bild darzustellen. Dabei werden verschiedene Einzelaufnahmen einer Kamerafahrt als schmale vertikale Streifen in einem Bildraum zusammengeführt, wobei jeder Streifen ein Kamerabild zu einem Zeitpunkt der Kamerafahrt abbildet. In der Folge entsteht eine bildraumbasierte multiperspektivische Darstellung. LORENZ und DÖLLNER greifen in [LORENZ & DÖLLNER 2009] das Prinzip der Streifenkamera auf, um Überblicksdarstellungen in einer interaktiven 3D-Anwendung zu realisieren. Im Unterschied zu einer Kamerafahrt werden in diesem Ansatz die Teilbilder gleichzeitig erzeugt und auf einer Bildebene zusammengeführt, wodurch eine Echtzeitfähigkeit gewährleistet ist.

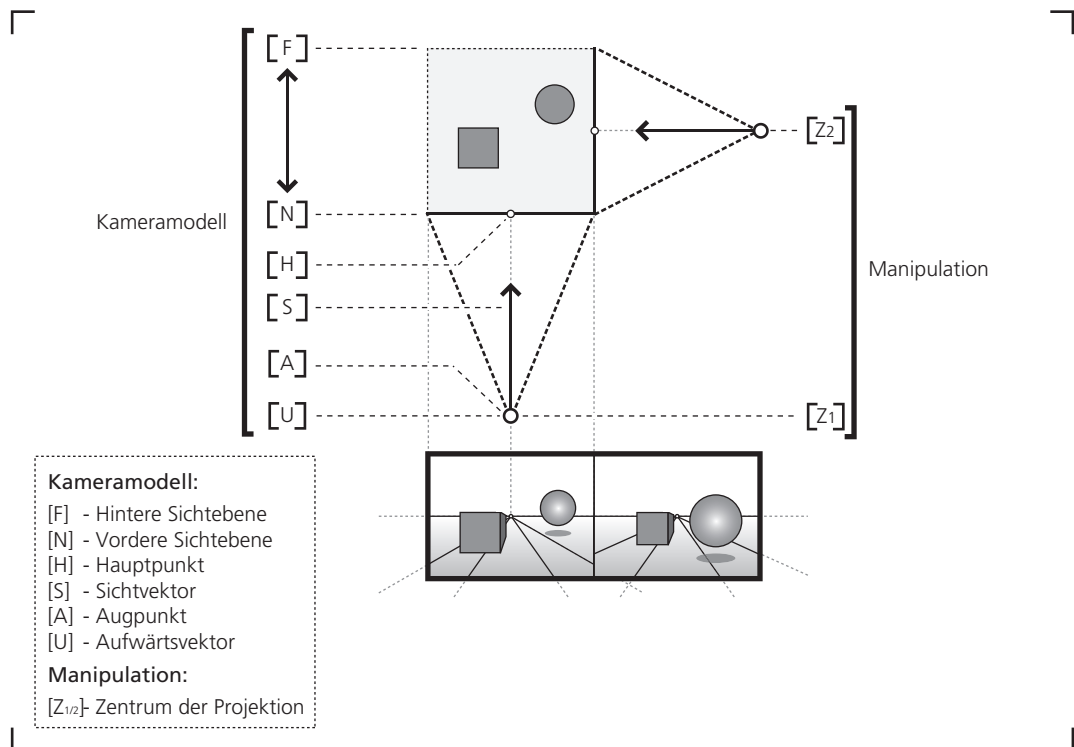


Abbildung 18: Kameramodell zur Erzeugung multipler Sichten auf Basis mehrere Projektionszentren [Z] und die durch Projektion entstehenden Abbilder des Objektraums

### 3.3.2.4 Manipulation am Zentrum der Projektion

Der konventionelle computergrafische Abbildungsprozess erzeugt einen Bildraum ausgehend von einem Augpunkt. Die Einbindung mehrerer Augpunkte in voneinander getrennten Bild-erzeugungsprozessen ermöglicht eine Visualisierung des Objektraums aus unterschiedlichen Einzelperspektiven. Im Unterschied zu einer multiperspektivischen Visualisierung werden die Einzelabbilder getrennt voneinander erzeugt, wodurch die Zusammenführung der Abbilder erst durch die Platzierung auf dem Ausgabegerät als *multiple Sichten* erfolgt (siehe Abbildung 18). Die Abbilder, die aus dieser Modifikation des computergrafischen Kameramodells hervorgehen, verfügen daher über keine inhärenten bildstrukturellen Zusammenhänge, bilden jedoch einen identischen Objektraum ab. Dadurch kann ein semantischer Zusammenhang zwischen den Sichten erstellt werden, indem die Montage von Abbildern die inhaltlichen Verbindungen einer visuellen Struktur mithilfe simultan dargebotener Sichten herstellt. Durch die Festlegung von Größe und Anordnung der Sichten wird ein Bild erzeugt, wodurch ein Betrachter die Einzelabbilder separat und gleichzeitig als Gesamtheit wahrnehmen kann. Diese Organisation der Elemente erzeugt ferner eine Dualität im Nutzer-Bild-Dialog. Einerseits wird durch mehrere Sichten eine Bewegung der Aufmerksamkeit induziert, da der Betrachter über die verschiedenen Abbilder gelenkt wird (vgl. [McCLOUD 1999, S. 78 ff.]). Andererseits wirken die multiplen Sichten einer ständigen Veränderung entgegen, indem die Abbilder in einer festen Anordnung platziert sind (vgl. [BORECZKY u. A. 2000]). Der Nutzen, der sich für den Anwender ergibt, beschreiben NORTH und SHNEIDERMAN folgendermaßen:

„visualizations with multiple coordinated views enable users to rapidly explore complex data and [...] relationships“ [NORTH & SHNEIDERMAN 2000, S. 128].

Eine Kopplung von Sichten, durch die Veränderungen in einer interaktiven Darstellung Auswirkungen auf andere Sichten haben, kann semantische Zusammenhänge zwischen den Visualisierungen zusätzlich verdeutlichen. In [PLUMLEE & WARE 2003A] werden relationale Sichten eingesetzt, um eine verbesserte Orientierung in komplexen räumlichen Darstellungen zu erreichen. Indes werden in [NORTH & SHNEIDERMAN 1997] koordinierte Sichten für die simultane Darstellung komplexer 3D-Daten genutzt. Der Einsatz koordinierter multipler Sichten in interaktiven Anwendungen ist allerdings mit einer Beschränkung des nutzerseitigen Interaktionsumfangs verbunden, da eine Sichtenkopplung die Veränderung in einer Visualisierung in Abhängigkeit zu einer Sicht und nicht zur Nutzerinteraktion stellt. Diese etwaige Einschränkung führt jedoch zu einer Erweiterung der Gestaltungsmöglichkeiten in Bezug auf die montierte Gesamtdarstellung. Die positiven Effekte, die sich aus der Koordination von Sichten ergeben, werden durch ROBERTS bekräftigt und deren Verwendung motiviert: *„Therefore, by utilizing a visualization design environment that enables the user to examine different representations and also manages their interactions and automatically coordinates operations between views, then they may perceive new and insightful relationships and facts from their data.“*<sup>48</sup> [ROBERTS 2007, S. 61]. In [WOJDZIAK U. A. 2011E] werden die Kopplungsmöglichkeiten auf Basis der computergrafischen Kamera für interaktive 3D-Anwendungen weiterführend beschrieben und deren Einsatz in [STARKE U. A. 2011, S. 70 ff.] als Mittel der Gestaltung anhand von Anwendungsszenarien aufgezeigt.

### 3.3.2.5 Manipulation am Sichtkörper

In Abschnitt 3.2.1 wird der Hauptpunkt bereits als ein bildstrukturelles Element eines zentralprojektiven Abbildes beschrieben. In der computergrafischen Fachliteratur wird dieser Durchstoßpunkt der optischen Achse mit der Bildebene nicht explizit identifiziert, sondern nur indirekt über die Definition des Kamerasichtkörpers beschrieben (vgl. [AKENINE-MÖLLER U. A. 2008]). Dies deutet darauf hin, dass die geometrische Mitte in konventionellen 3D-Anwendungen als gestalterisches Mittel bisher kaum eingesetzt wird. ARNHEIM veranschaulicht in [ARNHEIM 2003], dass die geometrische Mitte jedoch von entscheidender Bedeutung für die Ordnung eines Bildes ist und der visuellen Erschließung des Bildinhaltes dient. Die Verlagerung des Hauptpunktes induziert eine bildstrukturelle Veränderung, in deren Folge sich die Sichtbarkeit und die Anordnung der Elemente im Bildraum ändert, da durch die Position des Hauptpunktes die Lage des Horizontes und die Ausrichtung des Blickfeldes festgelegt sind. Die Modifikation des computergrafischen Sichtkörpers, die zu einer Verlagerung der geometrischen Mitte im Bild führt, wird in Abbildung 19 anhand des Kameramodells verdeutlicht. Die Abbildung zeigt einen asymmetrischen Sichtkörper, durch den sich der Hauptpunkt

<sup>48</sup> „Visualisierungen unter Verwendung von koordinierten multiplen Sichten ermöglichen dem Nutzer ein schnelles Untersuchen von komplexen Daten und Beziehungen“

<sup>49</sup> „Durch das Verwenden einer Visualisierungsumgebung, die das Untersuchen verschiedener Datenrepräsentationen ermöglicht und gleichzeitig die Interaktionen verwaltet und automatische Operationen zwischen den Ansichten koordiniert, wird dem Nutzer ermöglicht, neue, tiefgründige Zusammenhänge und Fakten aus seinen Daten zu gewinnen“

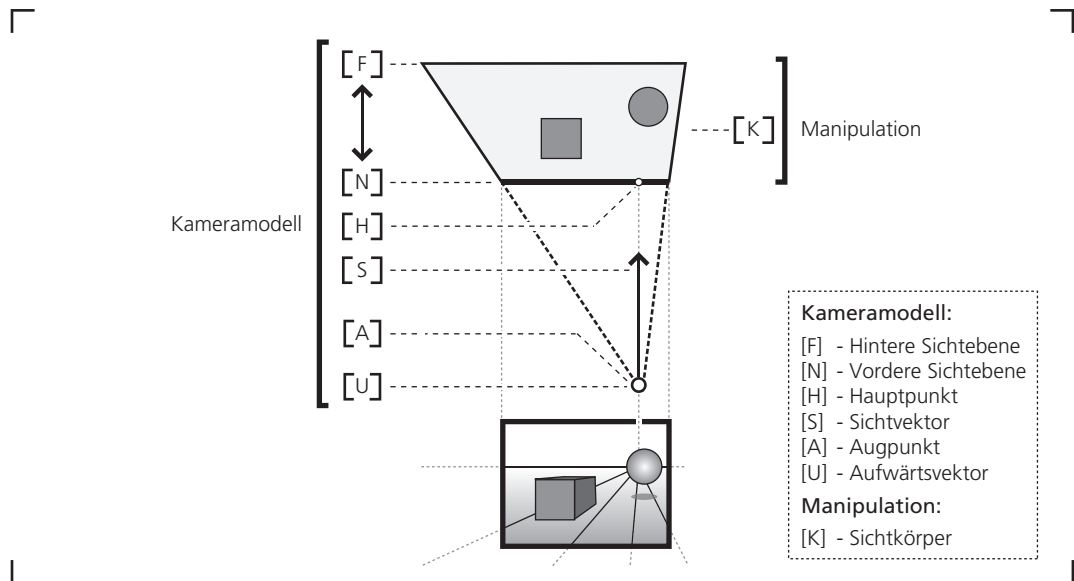


Abbildung 19: Kameramodell zur Verlagerung des Hauptpunktes mithilfe eines asymmetrischen Sichtkörper [K] und das durch die Projektion entstehende Abbild des Objektraums.

aufgrund eines geänderten Kamerasichtvektors in Bezug zur Bildebene, verlagert. FRANKE zeigt in [Franke 2006], dass die geometrische Mitte durch eine Sichtkörpermanipulation frei im Bildraum und über dessen Grenzen hinaus verortet werden kann. Diese freie Positionierung ermöglicht die Aufmerksamkeit des Nutzers gezielt zu lenken, da die Hauptpunktposition nach GROH den „Einstiegspunkt“ in ein Bild (vgl. [GROH 2007]) markiert und der Nutzer infolgedessen Auskunft über die Relevanz der Inhalte erhalten kann (vgl. [FRANKE U. A. 2008]). Eine dynamische Verortung der geometrischen Mitte in interaktiven 3D-Anwendungen bietet die Möglichkeit, eine Hauptpunktposition unabhängig von der Position des Bildes auf dem Ausgabegerät zu platzieren. Durch diese Trennung kann der sichtbare Ausschnitt des Objektraums angepasst werden, ohne eine Neuausrichtung der Kamera zu vollziehen, da die perspektivische Struktur des Abbildes der entkoppelten Position des Hauptpunktes weiterhin unterworfen ist.

Zusätzlich zu einer Entkopplung des Hauptpunktes von der Bildmitte kann durch die Manipulation des Sichtkörpers eine Neupositionierung des Hauptpunktes in der Bildraumtiefe erfolgen. Während bei der Zentralprojektion dieser Punkt scheinbar in der Tiefe des Bildraums positioniert ist, verortet sich der Hauptpunkt bei einer inversen Zentralprojektion vor dem Abbild. Der visuelle Eindruck entsteht, da die Objektkanten, der orthogonal zur Bildebene ausgerichteten Szenenelemente, in der Tiefe des Bildraums divergieren. LEVENE zeigt in [LEVENE 1998, S. 38–39], dass bei einer umgekehrten Perspektive Bereiche einer visuellen Struktur sichtbar werden, die in einer zentralprojektiven Darstellung, trotz eines identischen Augpunktes, nicht dargestellt sind. In der interaktiven Anwendung von BROSZ U. A. wird ein flexibles Sichtvolumen genutzt, um einen interaktiven Abbildungswechsel zwischen zentralprojektiver, parallelprojektiver und invers-zentralprojektiver Darstellung zu erreichen (vgl. [BROSZ U. A. 2007, S. 35–36]). Die Funktionsweise der Sichtvolumentransformation zur Erzeugung einer umgekehrten Perspektive ist in Abbildung 20 schematisch veranschaulicht.



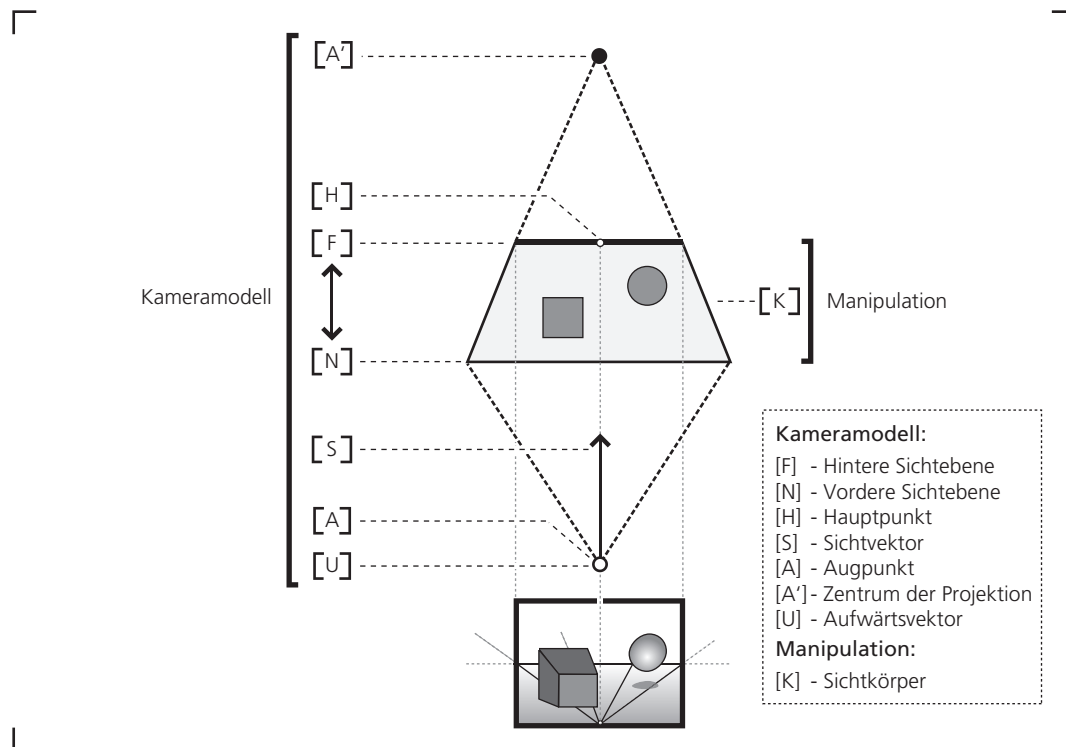


Abbildung 20: Kameramodell zur Erzeugung invers-zentralprojektiver Darstellungen durch einen invertierten Sichtkörper und das durch die Projektion entstehende Abbild des Objektraums.

Während bei einem zentralprojektiven Abbildungsvorgang die Position des Kameraaugpunktes mit dem Zentrum der Projektion identisch ist, wird die Zuordnung bei einer invers-zentralprojektiven Projektion aufgehoben. Der Kameraaugpunkt und das Zentrum der Projektion befinden sich auf entgegengesetzten Seiten der Bildebene (siehe Abbildung 20), wodurch die bildstrukturellen Eigenschaften eines linearperspektivischen Abbildes invertiert werden. Dieses charakteristische Merkmal führt nach FLORENSKIJ zu einer fixierten Anordnung der Objekte im Bildraum (vgl. [FLORENSKIJ 1997]). Die Relevanz der Inhalte kann nunmehr durch die Bewegung des Betrachters im Raum und von dessen Komposition der verschiedenen Sichten auf den Bildraum erschlossen werden [CRISTESCU 2009, S. 47]. Der Betrachter wird durch die Bildstruktur ferner dazu aufgefordert, sich die visuellen Strukturen durch Eigenbewegung zu erschließen (vgl. [CRISTESCU 2009, S. 47] nach [FLORENSKIJ 2000]).

### 3.3.3 Eine Systematisierung computergrafischer Projektionen

Die Analyse des computergrafischen Kameramodells sowie des Abbildungsprozesses zeigt eine weitaus größere Flexibilität in der Abbilderzeugung als derzeit als Mittel der Gestaltung für Benutzerschnittstellen von interaktiven 3D-Anwendungen eingesetzt wird. Die identifizierten bildstrukturellen Eigenschaften, die als ein Resultat aus der Analyse der verschiedenen computergrafischen Abbildungsvorgänge hervorgehen, fließen nunmehr in eine Einteilung von Darstellungsverfahren ein. Die nachfolgenden Eigenschaften bilden dabei die Grundlage der Systematisierung:

1. Die Anzahl der im Bild vorhandenen Hauptpunkte
2. Die Position der Hauptpunkte
3. Das geometrische Verhalten der orthogonal zur Bildebene verlaufenden Objektgeraden im Bildraum
4. Die Form der orthogonal zur Bildebene verlaufenden Objektgeraden im Bildraum
5. Der Datensatz, welcher dem Projektionsvorgang zugrunde liegt

Aufbauend auf diesen Merkmalen werden in Tabelle 1 die betrachteten Manipulationsmöglichkeiten reflektiert und die strukturellen Eigenschaften der resultierenden Bilder zusammengefasst. Basierend auf den benannten Merkmalen wird durch die konventionelle Zentralprojektion ein Abbild erzeugt, das durch einen Hauptpunkt charakterisiert ist (1), dessen Position sich in der Abbildmitte befindet (2). Die Objektgeraden, welche orthogonal zur Bildebene verlaufen, konvergieren geradlinig (4) in den Hauptpunkt (3), wobei die gesamte Szene in die Bildsynthese einbezogen ist (5). Die Weiteren herausgearbeiteten Bildräume unterscheiden sich hinsichtlich der Eigenschaften in einem oder mehreren Charakteristiken.

Basierend auf den identifizierten bildstrukturellen Eigenschaften kann eine Einteilung der Projektionsverfahren in lineare, kurvilineare und hybride Verfahren vorgenommen werden (siehe Tabelle 1). Zu den *linearen Projektionen* gehören die Verfahren der Zentral- und der Parallelprojektion sowie die in der 3D-Computergrafik kaum angewendete inverse Zentralprojektion. Die *kurvilinearen Projektionen* unterteilen sich in Fischaugen- und Panoramaprojektionen. Die *Multiprojektionen* umfassen Abbildungsvorgänge, die mehrere lineare Projektionsvorgänge in einem gemeinsamen Bildraum zusammenführen, und unterscheiden sich hinsichtlich der in den Projektionsschritt eingebundenen visuellen Strukturen, in »objektraumbasierte« und »bildraumbasierte Multiprojektion« unterschieden.

Die Einordnung der Abbildungsverfahren ist eine wichtige Voraussetzung das gestalterische Potenzial im Entwurf und in der Gestaltung interaktiver 3D-Anwendung gezielt einsetzen zu können. Durch die Einordnung ist eine Grundlage geschaffen, die aufzeigt, wie der Bildraum mithilfe eines Projektionsvorgangs verändert und als Mittel der Gestaltung eingesetzt werden kann. Weiterhin verdeutlichen die aufgezeigten unterschiedlichen bildstrukturellen Eigenschaften auf Basis des Projektionsschrittes eine Parametrisierbarkeit der Darstellungsmethoden, die einen flexiblen Einsatz in interaktiven 3D-Benutzerschnittstellen ermöglicht. Dadurch bietet sich für den Gestaltungs- und Entwurfsprozess die Möglichkeit die Gestalt von Elementen zu kontrollieren, die Anzahl an Elementen mit zunehmender Objektraumtiefe zu variieren oder mehrere Hauptpunkte zur Nutzerführung einzusetzen; ohne dabei den zugrunde liegenden Objektraum zu modifizieren.

Die Auswahl von Abbildungsverfahren im Entwicklungsprozess interaktiver Anwendungen stellt jedoch gleichzeitig eine hohe Herausforderung im Hinblick auf eine gebrauchstaugliche Interfacegestaltung dar. Daher ist es ein notwendiger und konsequenter Schritt auf Basis der gewonnenen Erkenntnisse eine Betrachtung der identifizierten Abbildungsverfahren im Hinblick auf deren Einbindung in interaktive 3D-Anwendungen zu vollziehen.

	Projektionsart	Anzahl der Hauptpunkte	Hauptpunktposition im Bildraum	Form der bildebenenorthogonalen Objektgeraden	Verhalten der bildebenenorthogonalen Objektgeraden	Genutzter Objektraum
Lineare Darstellungsform	Zentralprojektion	1	Bildmitte	Geradlinig	Konvergieren in den Hauptpunkt	Gesamt
	Parallelprojektion	0	-	Geradlinig	Verlaufen parallel	Gesamt
	Inverse Zentralprojektion	1	Bildmitte	Geradlinig	Divergieren ausgehend vom Hauptpunkt	Gesamt
Kurvilineare Darstellungsform	Panorama-projektion (zylindrisch)	2	Bildmitte und am Bildraumrand	Krummlinig (außer auf der Horizontlinie)	Konvergieren in einen Hauptpunkt	Gesamt
	Fischaugen-projektion	1	Bildmitte	Geradlinig	Konvergieren in den Hauptpunkt	Gesamt
Multi-projektive Darstellungsform	Objektraum-basierte Multiprojektion	>1	Variabel	Geradlinig	Konvergieren in einen Hauptpunkt	Teil
	Bildraumbasierte Multiprojektion	>1	Variabel	Geradlinig	Konvergieren in einen Hauptpunkt	Gesamt

Tabelle 1: Übersicht über die bildstrukturellen Eigenschaften von Bildräumen, die durch eine computergrafische Projektion auf Basis eines modifizierten Kameramodells erzeugt werden können.

### 3.4 Gestaltungshinweise für 3D-Interfaces

Der Entwurf von Benutzerschnittstellen interaktiver Systeme ist aufgrund der Vielgestaltigkeit nicht trivial, weswegen Richtlinien, Vorgaben und Standards die Dialoggestaltung unterstützen sollen. Die DIN EN ISO 9241 (vgl. [SCHNEIDER 2008]) ist ein internationaler Standard, der Richtlinien für die Kommunikation zwischen Mensch und Computer und mit dem Unterabschnitt 110 für die Benutzerschnittstellen von interaktiven Systemen beschreibt. Weiterhin werden in der Literatur grundlegende Richtlinien beschrieben, beispielsweise die Gestaltungsregeln von SHNEIDERMAN (vgl. [SHNEIDERMAN 2004]), die sich als Quasistandard etabliert haben. Darüber hinaus existieren Grundregeln der Gestaltung, die wesentliche und allgemeingültige Aspekte des Interfacedesigns aufzeigen (vgl. [STAPELKAMP 2007]).

Zur Vermittlung eines allgemeinen Verständnisses für den strukturellen Aufbau und die bildliche Ausgestaltung von 3D-Interfaces werden Gestaltungshinweise ausgearbeitet. Diese bieten Hinweise und Empfehlungen zur Gestaltung interaktiver 3D-Anwendungen. Dabei definieren diese keine formalen Regeln, sondern sind mit dem Ziel formuliert, Hilfestellungen für das Design interaktiver 3D-Benutzerschnittstellen anzubieten. Obgleich Gestaltungshinweise im Allgemeinen kein Garant für ein gutes Interface- oder Interaktionsdesign einer Benutzerschnittstelle sind und nicht zwingend, wie NIELSEN in [NIELSEN 1994, S. 155] betont, die Gebrauchstauglichkeit einer Benutzerschnittstelle sicherstellen, bieten diese ein fundiertes

Wissen für die Entwicklung zukünftiger Benutzerschnittstellen. Der Schwerpunkt der Betrachtung liegt auf dem Entwurf einzelner 3D-Darstellungen sowie dem koordinierten Einsatz interaktiver, computergrafischer Abbilder unter Verwendung von unterschiedlichen Visualisierungstechniken.

### 3.4.1 Akzentuierung in der Darstellung

Für die Visualisierung eines Objektraumes auf einer planaren Projektionsfläche ist es erforderlich, die technischen Rahmenbedingungen des Abbildungsprozesses zu berücksichtigen und diese Gegebenheiten in den Gestaltungsprozess einzubeziehen:

- Durch das Ausgabegerät ist das Abbild flächig. Bei stereoskopischen und autostereoskopischen Darstellungen wird diese Eigenschaft aufgelöst, jedoch ist derzeit die überwiegende Anzahl an Ausgabegeräten für interaktive Anwendungen nicht mit einer stereoskopischen Darstellungstechnologie versehen. Darüber hinaus ist eine äquivalente Nutzung stereoskopischer Ausgabegeräte bisher nicht abschließend erforscht.
- Das computergrafische Abbild ist durch die Sichtkörperdefinition der computergrafischen Kamera begrenzt und wird durch den für die Darstellung zur Verfügung stehenden Bereich auf dem Ausgabegerät gerahmt. Dies führt dazu, dass jedes Abbild durch ein Anwendungsfenster oder die Begrenzungen des Ausgabegerätes limitiert ist.
- Jede Projektion einer räumlichen Szene auf einem Ausgabegerät erfolgt auf Basis einer festgelegten Kameraposition und Ausrichtung. Dadurch entsteht das Abbild aus einer definierten Perspektive und unterliegt den Gesetzen des angewendeten Projektionsverfahrens.

Die Rahmenbedingungen zeigen, dass ein computergrafisches Abbild in dessen möglichen Ausprägungen durch die Gegebenheiten des Ausgabemediums begrenzt wird. Eine etwaige Konsequenz zur Überwindung dieser Limitierung und zur Erreichung einer möglichst großen Darstellungsvielfalt ist die technische Überwindung der Restriktionen durch erweiterte visuelle Ausgabegeräte, beispielsweise das Head-Mounted Display oder die CAVE. Diese Technologien können die Auflösung der Beschränkung jedoch nur in Teilen erreichen. Vielmehr ist der intendierte Einsatz von bildgebenden Techniken ein aussichtsreicher Ansatz (vgl. [DURAND 2002b, S. 28ff.]), um Benutzerschnittstellen zu gestalten und die Limitierungen des Mediums zu kompensieren. Darüber hinaus ist das Akzentuieren einer Benutzerschnittstelle möglich, um eine intendierte Bildaussage zu integrieren. Dafür ist es notwendig, das Streben nach dem Abbild der Realität zurückzustellen und eine Hinwendung zu generalisierten computergrafischen Abbildern zu forcieren. Für die Gestaltung von Benutzerschnittstellen ist es erforderlich der Limitierung, unter Zuhilfenahme des gezielten Einsatzes von Projektionsverfahren, Farben, Texturen und Beleuchtung, entgegenzuwirken. Es ist beispielsweise durch nicht-fotorealistische Visualisierungen möglich die Übertragung von Tiefeninformationen in planaren Abbildern durch den Einsatz von Strichtexturen zu unterstützen (vgl. [MEIER 1996]), die Darstellung von Informationen durch eine gezielte Beleuchtung des Objektraums zu begünstigen (vgl. [GOOCH u. A. 1998]) oder die Perspektivität den Szenen- oder Kontextgegebenheiten dynamisch anzupassen (vgl. [BROSZ u. A. 2007]). Durch die große Anzahl an bereits existierenden Techniken steht ein Potenzial an Gestaltungsmöglichkeiten zur Verfügung, das

im konkreten Einsatz eine Expressivität, eine Klarheit und einen Ausdruck unter Berücksichtigung der technischen Gegebenheiten bewirkt. Dadurch kann dem Anwender ein für das Ausgabemedium optimiertes Abbild des Objektraums dargeboten werden.

### 3.4.2 Einsatz linearer und nichtlinearer Darstellungsformen

Die Zentralprojektion ist das konventionelle Abbildungsverfahren in der computergrafischen Bildsynthese. Dadurch suggerieren bestehende 3D-Anwendungen, dass alleinig durch lineare Projektionsverfahren die Abbildung des computergrafischen Objektraums erreicht werden kann. Die Betrachtung bildstruktureller Eigenschaften in der bildenden Kunst hat bereits aufgezeigt, dass es eine Reihe weiterer Abbildungsmöglichkeiten gibt, deren Eigenschaften situationsabhängig einen Zuwachs an der zu betrachtenden Informationsmenge ermöglicht oder der Verständlichkeit von Informationen verbessert. Zudem wird in [STROTHOTTE & SCHLECHTWEIG 2002, S. 10–11] auf die Notwendigkeit hingewiesen, die Möglichkeiten zur Verbildlichung des virtuellen Raumes zu nutzen und die computergrafische Projektion nicht ausschließlich unter naturalistischen Zielsetzungen anzuwenden. Weiterhin sprechen sich unter anderem SINGH in [SINGH 2002, S. 17–18] und AGRAWALA U. A. in [AGRAWALA U. A. 2000, S. 125–126] gegen die ausschließliche Schaffung linearer Abbilder und für den Einsatz der Nichtlinearität in der Gestaltung von Benutzerschnittstellen aus. Die dafür bereitstehenden computergrafischen Abbildungsverfahren werden jedoch aufgrund mangelnder Kenntnis oder Vertrautheit, kaum eingesetzt.

Nichtlineare Abbilder sollten verstärkt in 3D-Anwendungen integriert werden, da sie die Informationsdarbietung verbessern und dadurch die Gebrauchstauglichkeit einer Benutzerschnittstelle erhöhen können. Ferner dienen die Darstellungstechniken als Lösungsansatz, um die Nachteile einer zentralprojektiven Darstellung (siehe Unterabschnitt 3.1.4) zu beseitigen. Durch eine aktive Einbindung der bildstrukturellen Eigenschaften als Mittel der Gestaltung sollten Darstellungsweisen, die zu Bildern mit einem flächigen Charakter führen, der simultanen Präsentation einer Szene dient. Dagegen begünstigen die Visualisierungen mit einer starken Räumlichkeit das Erschließen von Entfernung und Tiefe und sollten zur Erfassung der visuellen Strukturen sukzessiv dargeboten werden. Bereits im Gestaltungs- und Entwurfsprozess interaktiver 3D-Anwendungen ist es notwendig dabei die unterschiedlichen Projektionsverfahren zu berücksichtigen, um dem Nutzer alternative Bildräume kontext- und aufgabenspezifisch bereitzustellen. Durch eine Beschreibung von Visualisierungstechniken, in der Parameter und Funktionalität im Hinblick auf das Visualisierungsziel zur Verfügung stehen, wird die Nutzung der Projektionsverfahren in einer interaktiven Anwendung gefördert und die gezielte Einbindung in das Interfacedesign vereinfacht. Die Anwendbarkeit der Darstellungstechniken sollte weiterhin begünstigt werden, indem die Verfahren als voneinander unabhängige Werkzeuge für die Abbilderzeugung zur Verfügung stehen. Dadurch wird ein flexibler und anwendungsunabhängiger Einsatz der Verfahren gewährleistet. Ein unnötiger oder ein falscher Einsatz kann indes aufgrund der teilweise komplexen bildstrukturellen Eigenschaften zu einer unnötigen kognitiven Belastung des Anwenders führen und diesen vom tatsächlichen Kommunikationsziel ablenken. Dies kann zu einer Unzufriedenheit beim Nutzer und gleichzeitig zu einer unnötigen Beanspruchung der Systemressourcen führen.

### 3.4.3 Koordination multipler 3D-Darstellungen

Wenn ein gegebener Datenbestand in der Überführung in eine visuelle Struktur eine höhere Komplexität aufweist, als es eine einzelne Abbildung vermag effizient zu veranschaulichen, ist es notwendig, die Darstellung der Informationen auf mehrere Abbilder zu verteilen. Situativ ist es zweckdienlich unterschiedliche Merkmale oder Wertebereiche eines Datenbestandes aufzeigen zu können oder aber die Vergleichbarkeit von Merkmalen eines Datenbestandes zu gewährleisten. Die Informationen sollten dabei nicht sukzessiv, sondern simultan dargestellt werden, auch wenn die darzustellende Fläche für das einzelne Element in der Folge reduziert werden muss. In einem 3D-Interface sollten hierbei multiple Sichten das Interface bilden, wobei die Trennung der Einzelabbilder vergleichbar zum Aufbau eines Comics erfolgen sollte. Durch eine gerahmte Abbildfläche, das *Panel*, wird eine Abgrenzung zwischen den Visualisierungen erreicht. Die einzelnen Sichten sollten weiterhin durch ein Gutter<sup>50</sup> strukturiert werden. Die entstehende Anordnung und Lesereihenfolge sowie der Freiraum zwischen den Abbildern nutzt der Anwender, um den zeitlichen oder gegenständlichen Übergang zwischen den Panels herzustellen und infolgedessen den semantischen Zusammenhang zu erkennen (vgl. [McCLOUD 1999, S. 78 ff.]). In der Arbeit von BALDONADO ([WANG BALDONADO u. A. 2000]) werden Regeln für die Gestaltung von multiplen Sichten im Bereich der Informationsvisualisierung formuliert. BALDONADO zeigt in der Arbeit Aspekte auf, die ebenfalls für multiple Sichten eines 3D-Interfaces gelten und in der Folge zu berücksichtigen sind. Es sollte darauf geachtet werden, dass die Komplexität eines Datenbestandes nur verringert werden kann, indem mehrere Sichten auf einen Datenbestand die Disparitäten und Zusammenhänge erkennbar werden lassen. Gleichzeitig muss sichergestellt sein, dass die Sichten verschiedene Attribute, Modelle oder Abstraktionslevels des Datenbestandes visualisieren. Die Nutzung von visuellen Wahrnehmungshinweisen zur Verdeutlichung von Beziehungen zwischen mehreren Sichten sollte für den Nutzer dabei jederzeit kenntlich gemacht werden. Bei komplexen Datenbeständen kann die Unterteilung in verständliche Segmente erreicht werden, indem die Verknüpfung dieser Segmente über die Dimensionen des Datenbestandes oder über die Abbildungsparameter der Sichterzeugung erfolgt. In Abhängigkeit von der darzustellenden Information und der zu lösenden Aufgabe ist eine Verknüpfung notwendig, um den Interaktionsaufwand und den kognitiven Aufwand des Betrachters zu reduzieren.

Durch eine Koordination von Sichten kann ein zusätzlicher Nutzen für den Nutzer-Bild-Dialog entstehen, der bereits in 2D-Benutzerschnittstellen gewinnbringend eingesetzt wird (vgl. [FELDT u. A. 2005, S. 139–140]). Im Kontext von 3D-Visualisierungen wird in [MAPLE u. A. 2004, S. 781 ff.] gezeigt, dass ein Einsatz von multiplen Sichten in 3D-Visualisierungen ebenfalls Vorteile für die Informationserfassung bietet. Die Studien von PLUMLEE und WARE in [PLUMLEE & WARE 2003b, S. 198–199] bestätigen dies. ANDRIENKO und ANDRIENKO stellen in [ANDRIENKO & ANDRIENKO 2007] den generellen Nutzen von koordinierten multiplen Sichten heraus, indes ROBERTS darauf hinweist, dass der Verwendung in interaktiven Anwendungen nur durch eine nutzerzentrierte Bereitstellung der Koordinierungsfunktionalität erfolgen sollte (vgl. [ROBERTS 2007, S. 68–69]).

<sup>50</sup> *Gutter* engl. Abflussrinne, Spaltenzwischenraum

### 3.4.4 Gewährleistung der Echtzeitfähigkeit

Der Einsatz kurvilinear und multiprojektiver Abbildungsverfahren bietet eine Vielgestaltigkeit bei der Erzeugung computergrafischer 3D-Darstellungen. Gleichzeitig sind jedoch die Verfahren, die zur Erzeugung der nichtlinearen Visualisierungen eingesetzt werden, beispielsweise krummlinige Darstellungen auf Basis des Raytracing-Verfahrens, rechenaufwändiger als lineare Projektionsverfahren, deren Berechnung innerhalb der Bildsynthese optimiert ist. Beim Einsatz nichtlinearer Abbildungsverfahren ist darauf zu achten, dass bei deren Anwendung die gebotene *Bildwiederholfrequenz* den interaktiven Nutzer-Bild-Dialog nicht gefährdet beziehungsweise stört. Daher ist es notwendig eine 3D-Benutzerschnittstelle immer unter Berücksichtigung der zur Verfügung stehenden Hard- und Software zu entwickeln. Dies umfasst die Anzahl der dargebotenen Sichten, die Menge der zur Bilderzeugung eingesetzten Kameras und die verwendenden Abbildungsverfahren. Da eine angestrebte Bildwiederholfrequenz darüber hinaus vom darzustellenden Objektraum abhängt, kann eine Sicherstellung der Echtzeitfähigkeit vor allem durch den zielgerichteten Einsatz nichtlinearer und multiprojektiver Abbildungsverfahren sowie durch eine Überprüfung der *Echtzeitfähigkeit* zur Laufzeit einer Anwendung auf verschiedenen Testsystemen mit realen, kontextrelevanten Daten sichergestellt werden.

### 3.4.5 Der interaktive Prozess mit 3D-Benutzerschnittstellen

Der Ausschluss des Nutzers vom Vorgang der Bilderzeugung kann dazu führen, dass der Anwender in seinen Bedürfnissen und Anforderungen übergangen wird. In der Folge sinkt für diesen die Akzeptanz einer interaktiven Anwendung als unterstützendes Werkzeug. Darin zeigt sich eine Wechselwirkung zwischen der Problemkenntnis eines Nutzers und dem Automatisierungsgrad einer Anwendung. Es ist zu konstatieren, dass weder ein vollständig manueller noch ein gänzlich automatischer Interaktionsprozess in 3D-Anwendungen empfehlenswert ist. Der Kommunikationsprozess sollte vielmehr in einem interaktiven Dialog zwischen Nutzer und System vollzogen werden. Dabei muss der Nutzer im Erreichen seiner Ziele durch systemseitige Teilautomatismen unterstützt werden, indem dieser durch das System befähigt wird, wesentliche Automatismen zu initiieren und zu steuern. Dabei sollte die konkrete Ausgestaltung des Dialogverlaufes vom Anwender ausgehen. Dadurch wird sichergestellt, dass die erzeugten Darstellungen in Relation zum Nutzerziel stehen und der Grad der Automatisierung durch den Anwender festgelegt ist. Es muss weiterhin gewährleistet sein, dass die Benutzerschnittstelle eine Darstellung in angemessener Zeit (möglichst in Echtzeit) aktualisiert, um im interaktiven Nutzer-Bild-Dialog auf nutzer- und systemseitige Änderung zu reagieren. Die Abbildungserzeugung sollte dabei immer unter Berücksichtigung des Ausgabemediums erfolgen und an den Kenntnisstand und die Fähigkeiten des Nutzers angepasst sein, um die Gebrauchstauglichkeit einer Darstellung zu erhöhen.

Ein effektiver Nutzer-Bild-Dialog kann umgesetzt werden, indem im Entwurf einer Benutzerschnittstelle die elementaren Kontextfaktoren eines Nutzungsziels identifiziert und als Anforderungen in den systemseitigen Bilderzeugungsprozess eingebunden werden. Diese Faktoren können in notwendige (beispielsweise eine verzerrungsfreie Darstellung von Objekten für relationale Aufgaben) und gewünschte Bedingungen (beispielsweise die Ausrichtung von



Elementen für eine objektbezogene Aufgabe) unterschieden werden. Es sollten die relevanten Darstellungsmöglichkeiten durch Entscheidungshilfen, in Form von vordefinierten Änderungsmodi für den Anwender verständlich und kontextbezogen zur Verfügung gestellt werden. Die Verwaltung der Balance zwischen manueller Kontrolle des Nutzers und automatischen Ausführungen des Systems sieht DURAND als einen Schlüssel für einen effizienten Nutzer-Bild-Dialog in Bezug auf computergrafische Darstellungen [DURAND 2002A, S. 113–114]. LANGE zeigt in [LANGE U. A. 2006, S. 7 ff.], dass aufgrund der Kontextabhängigkeit im Visualisierungsprozess ein interaktives Visualisierungsdesign anzuwenden ist. Dadurch kann das zu erzeugende Abbild hinsichtlich konkreter Daten, Ziele, Ressourcen und Nutzereigenschaften spezifiziert werden. AGRAWALA U. A. zeigen in [AGRAWALA U. A. 2000, S. 130 ff.] eine interaktive Anwendung, die unter Berücksichtigung von nutzerdefinierten Kontextfaktoren eine semiautomatische Bilderstellung unterstützt. Ferner veranschaulichen AGRAWALA und STOLTE in einem konkreten Anwendungsszenario die Anwendbarkeit von Teilautomatismen in Kombination mit Nutzerwünschen und Vorgaben für eine effektive Routendarstellung für Navigationssysteme (vgl. [AGRAWALA & STOLTE 2001]).

### 3.4.6 Interdisziplinarität im Entwicklungsprozess interaktiver 3D-Anwendungen

Bei aktuellen interaktiven 3D-Anwendungen werden hohe Anforderungen an die Benutzerschnittstelle in Bezug auf deren Gebrauchstauglichkeit gestellt. Um diese zu erfüllen, ist es notwendig, Softwareentwickler, Interaktionsdesigner und Experten aus dem jeweiligen Anwendungsfeld in einem kooperativen Entwurfs- und Entwicklungsprozess zusammenzuführen. Dadurch können deren Kompetenzen im Entwurfsprozess gebündelt werden. Die unterschiedlichen Personengruppen werden aufgrund ihrer ergänzenden Fähigkeiten in der Anwendungsentwicklung benötigt, nutzen jedoch unterschiedliche Vokabulare, die durch das jeweilige domänenspezifische Wissen geprägt sind. Die unterschiedlichen Betrachtungsweisen und der unterschiedliche Kenntnisstand sollten für den Entwurf von 3D-Benutzerschnittstellen auf einer gemeinsamen methodischen Basis zusammengeführt werden, wobei die Fähigkeiten und Gewohnheiten der beteiligten Akteure zu berücksichtigen sind. Die Identifikation von Anwendungsszenarien, die Beschreibung von Anwendungsfällen, die Festlegung eines Storyboards<sup>51</sup> sowie eine prototypische Umsetzung sind wesentliche Schritte in einem Entwurfsprozess, die im Entwurf von 3D-Benutzerschnittstellen interdisziplinär ausgeführt werden sollten (vgl. [MEMMEL & REITERER 2008, S. 359–360]). Diese Prozessschritte und die Ergebnisse aus diesen Vorgängen sollten durch eine Auswahl geeigneter und verbundener Sprachen und Konzepte fusioniert werden. Ein konsequenter Schritt für den Entwurf von 3D-Benutzerschnittstellen ist die Verwendung einer gemeinsamen Beschreibungsform, wie sie in [BORCHERS 2000b, S. 114] als eine „*lingua franca*“ innerhalb interdisziplinärer Teams gefordert wird. Die Beschreibungsform sollte die Methoden, die Erfahrungen und die Werte über eine Software-Architektur den Interaktionsdesignern und Anwendungsexperten verdeutlicht können. Ebenso müssen Gestaltungsrichtlinien, Standards, Regeln und Vorgaben in dieser gemeinsamen Sprache ausgedrückt werden können. Dies bietet für eine

<sup>51</sup> *Storyboard* engl. Szenenbuch; ist eine (sequenzielle) Bilderfolge und dient im Interfaceentwurf der Visualisierung eines Konzeptes oder eines Interaktionsdesigns

Kommunikation eine fundierte Basis von Methoden, Vorgehensweisen und Entwurfsentscheidungen, wodurch beispielsweise Konzepte mit einem einzigen Wort veranschaulicht und kommuniziert werden können. Gleichzeitig muss das Vorgehen im Entwurfsprozess auf Basis der Beschreibungsform für alle beteiligten Personen transparent und nachvollziehbar sein. Eine praktische Verwirklichung einer interdisziplinären Entwurfsmethodik zeigen WINGRAVE und BOWMAN in [WINGRAVE & BOWMAN 2008] anhand der Beschreibung von 3D-Interaktionen für virtuelle Umgebungen in verschiedenen Abstraktionsebenen, wodurch die Interaktionen für verschiedene Personengruppen nachvollziehbar werden. Eine Umsetzung eines interdisziplinären Entwurfswerkzeuges für die Spezifikation von 2D-Interfaces wird in [MEMMEL & REITERER 2008, S. 361 ff.] vorgestellt. ALEXANDER verwendet den Ansatz einer gemeinsamen Beschreibungsform in [ALEXANDER U. A. 1977], um Entwurfsmuster zu entwickeln, die es Architekten und Laien ermöglicht Architekturentwürfe zu entwickeln. Die Muster, die durch Beschreibungen und Beispiele ein allgemeines Verständnis für Entwürfe im Bereich der Architektur bereitstellt, bilden in der Zusammenführung eine Mustersprache, die domänenübergreifend Anwendung findet.

### 3.4.7 Konklusion der Gestaltungshinweise

Die Umsetzung der aufgestellten Gestaltungsrichtlinien erfordert Fachwissen im Bereich der Informatik aber auch in den Bereichen des Interfacedesigns und der Bildsprache. Neben der Anwendung von elementaren Gestaltungsregeln und -prinzipien kann durch eine interdisziplinäre Anwendung der Gestaltungshinweise ein Benutzerschnittstellenentwurf durch die Einbindung bestehender Methoden und neuer Techniken profitieren. Nachfolgend werden die vorgestellten Gestaltungshinweise für interaktive 3D-Anwendungen in kompakter Form zusammengefasst.

- Verwende bildgebende Verfahren als ein Werkzeug der Kompensation und der Akzentuierung, um die Restriktionen eines planaren Ausgabemediums zu überwinden.
- Wende neben den linearen Projektionen ebenso die nichtlinearen Abbildungsverfahren an, um kontextbezogene und nutzerspezifische Bildräume im Nutzer-Bild-Dialog zu erzeugen.
- Setze koordinierte multiple Sichten für eine verteilte Visualisierung einer konzeptionellen Entität ein, wenn eine Aufteilung der komplexen Informationen in einer reorganisierten Darstellungsweise der besseren Erfassung von Informationen dient.
- Verwende eine modifizierte Bildsynthese in einer Form, die eine Echtzeitfähigkeit der Anwendung nicht gefährdet.
- Etabliere einen interaktiven Prozess, indem der Nutzer durch Teilautomatismen bei der Erreichung seiner Ziele unterstützt wird, der Grad der Automatisierung jedoch durch diesen festgelegt werden kann.
- Nutze eine Sprache, Methode oder Technik, die eine Ausdrucksform im Entwurfsprozess bietet und die von Softwareentwicklern, Designer und Domänenexperten gleichermaßen verstanden und genutzt werden kann.

### 3.5 Systematisierung von 3D-Abbildungsverfahren

Interaktive 3D-Systeme sind durch zwei wesentliche Merkmale charakterisiert – die bildliche Repräsentation einer dreidimensionalen Szene und die bildgestützte Interaktion des Nutzers mit dem Interface. Während sich die visuelle Repräsentation durch die computergrafischen Darstellungen auf dem Ausgabegerät auszeichnet, beschreibt die Interaktion die Kommunikation zwischen Anwender und System. Der Dialog dient dazu den Nutzer unter anderem in die Lage zu versetzen, Informationen zu extrahieren, lokale Bereiche zu vergleichen oder einen Gesamtüberblick über eine Gegebenheit zu erhalten. Die Verwirklichung von effektiven computergrafischen Darstellungen ist kontextabhängig, wobei eine Vielzahl von Faktoren darüber entscheidet, inwieweit das kommunikative Ziel, das mit der visuellen Repräsentation verbunden ist, tatsächlich erreicht wird. Sowohl die Betrachtung der bildstrukturellen Eigenschaften einer Visualisierung, die durch ein Abbildungsverfahren festgelegt werden (siehe Unterabschnitt 3.3.3), als auch die mit einer Darstellung verfolgten Ziele führen zu einem besseren Verständnis für den effektiven Einsatz von Visualisierungen in 3D-Anwendungen.

#### Verwandte Klassifikationen

In der Literatur liegen, hinsichtlich des Kommunikationsprozesses zwischen einem Nutzer und einem interaktiven System, eine Vielzahl von Problembeschreibungen, Zielsetzungen und möglichen Dialogtechniken vor, die in Teilen erhebliche inhaltliche Überschneidungen aufweisen. Daher werden nachfolgend Klassifikationen betrachtet, die anwendungsunabhängige und allgemeine Anforderungen sowie Ziele des Nutzer-Bild-Dialogs formulieren. Anwendungsabhängige, spezifische Aspekte werden durch den allgemeineren Begriffsraum implizit beschrieben. Trotz großer Überschneidungen besitzen die Taxonomien unterschiedliche Abstraktionsgrade und Kategorisierungsansätze. Während einige eine Ordnung in Bezug auf die Dialogtechnik aufbauen, fokussieren andere die Aufgaben und Ziele des Nutzers als Ausgangspunkt der Systematisierung.

In [ROTH & MATTIS 1990, S. 199] werden verschiedene Ziele bei der Suche nach Informationen, beispielsweise Werte suchen und vergleichen, Verteilungen identifizieren sowie Korrelationen finden, identifiziert. Durch die Systematisierung von ROTH und MATTIS soll es Designern und Entwicklern möglich sein, Visualisierungen an die Bedürfnisse von Nutzergruppen anzupassen. Aufbauend auf diesen Ausführungen trennen WEHREND und LEWIS in [WEHREND & LEWIS 1990, S. 140 ff.] die Ziele der Mensch-Computer-Interaktion in *Objekt* (beispielsweise die Eigenschaften der Daten) und *Operation* (beispielsweise identifizieren und vergleichen). Robertson führt darüber hinaus in [ROBERTSON 1990, S. 117] den Ortsbezug als zusätzlichen Aspekt in die Systematisierung von Zielen im Nutzer-Bild-Dialog ein. Dies ermöglicht eine Festlegung, inwieweit der Anwender globale, lokale oder punktuelle Informationen zur konkreten Zielerreichung fokussiert. In [KELLER & KELLER 1993] werden aufbauend auf der Systematisierungsgrundlage von ROBERTSON unterschiedliche praxisnahe Anwendungsbeispiele zur Erreichung von Visualisierungszielen unter Einbeziehung von Datentypen, Operationen, Objekten und dem Bezugsraum präsentiert. In Anlehnung an [WEHREND & LEWIS 1990] werden in [FUJISHIRO U. A. 2000, S. 3] Visualisierungsziele beschrieben und von den Aktionen, die zur Erreichung der Ziele notwendig sind, separiert. FUJISHIRO U. A. zeigen in dieser Arbeit, dass

diese Trennung eine Kombination von beliebigen Zielen und Aktionen erlaubt. Zusätzlich werden die zwei Kategorien mit den von SHNEIDERMAN definierten Zielen kombiniert (vgl. [FUJISHIRO U. A. 2000, S. 2–3]). In dessen *Task-by-Data-Type-Taxonomy* [SHNEIDERMAN 1996, S. 337 ff.] führt SHNEIDERMAN verschiedene Aufgaben (beispielsweise Überblick, Zoom und Filtern) und Datentypen (beispielsweise 1D, 2D und Hierarchie) ein, um Entwurfsvorlagen für eine effektive Darstellung von Informationen im Interaktionsprozess bereitzustellen. In [NOCKE & SCHUMANN 2004, S. 2] wird ebenfalls eine verallgemeinerte Spezifikation von Zielen und Aufgaben zur Unterstützung des Visualisierungsprozesses dargelegt, welche die Nutzerintention durch eine Attributierung des Visualisierungsziels beschreibt und deren Komposition zur Festlegung komplexer Ziele eingesetzt werden kann (vgl. [NOCKE & SCHUMANN 2004, S. 3-4]). ANDRIENKO und ANDRIENKO formulieren in [ANDRIENKO & ANDRIENKO 2006, S. KAPITEL 3], unter Nutzung einer formalen Notation, anwenderbezogene Visualisierungsziele und beziehen sowohl elementare als auch zusammengesetzte Zielsetzungen ein. Die Bestrebungen von ANDRIENKO und ANDRIENKO fokussieren dabei die Schaffung eines Verständnisses für Visualisierungsziele und die Nutzung von Visualisierungsmethoden mithilfe von Werkzeugen, basierend auf den Erkenntnissen bestehender Systematisierungen und Klassifikationen.

### 3.5.1 Eine aufgabenorientierte Systematisierung

Für eine Systematisierung von Darstellungsverfahren im Kontext eines aufgaben- und zielbezogenen, interaktiven Nutzer-Bild-Dialogs bieten sich zahlreiche Kriterien an. Da die Formen computergrafischer Bildräume unterschiedliche Eigenschaften besitzen und deren Erzeugung verschiedene Rahmenbedingungen aufweisen, ist eine allgemeingültige und umfassende Klassifikation nicht trivial. Im Hinblick auf die tatsächliche Verwendung einer Systematisierung ist zunächst die Nutzergruppe zu identifizieren, die von dieser Klassifikation Gebrauch machen soll. Die nachfolgende Systematisierung richtet sich primär an 3D-Interfacedesigner und -entwickler, um diese Nutzergruppe im Entwurfsprozess zu unterstützen. In der Folge wird diese einen pragmatischen Charakter besitzen, da eine sehr informationstheoretische oder mathematische Taxonomie für einen Designer oder einen Softwareentwickler wenig hilfreich wäre. Damit unterscheidet sich dieser Ansatz von Klassifikationen für computergrafische Abbildungsverfahren, die beispielsweise eine mathematische Beschreibung (vgl. [SALOMON 2006]) in den Vordergrund stellen und ist eher mit der Systematisierung von Interaktionswerkzeugen in [DACHSELT 2004, S. 57ff.] vergleichbar. Zunächst werden mögliche Kriterien, die als Grundlage für eine Klassifikation dienen können, identifiziert und deren Anwendbarkeit in Bezug auf Ordnung und Abgrenzung beurteilt:

- *Das Anwendungsgebiet:* Hierbei handelt es sich um ein sehr allgemeines Kriterium, das aufgrund der Vielfalt von Disziplinen und Domänen ungeeignet ist und Mehrfacheinordnungen zur Folge hätte.
- *Der Manipulationsgegenstand:* Dieses Kriterium umfasst die Manipulationsmöglichkeiten, welche die computergrafische Kamera bereitstellt (siehe Unterabschnitt 3.3.2). Da jedoch durch verschiedene Modifikationsmöglichkeiten gleichartige Bildräume erzeugt werden können, ist dieses Kriterium für eine Klassifikation ungeeignet. Beispielsweise

kann eine Panoramadarstellung durch eine bildraumbasierte multiprojektive Abbildung und durch eine kurvilineare Projektion erreicht werden.

- *Die Darstellungsform:* Die verschiedenen bildstrukturellen Eigenschaften, beispielsweise die Anzahl der Hauptpunkte im Bild oder die Form der Objektgeraden im Bildraum (siehe Unterabschnitt 3.3.3) können zur Klassifikation herangezogen werden. Die Einteilung bietet jedoch eine theoretische und weniger eine pragmatische Strukturierung.
- *Der Wirkungsbereich:* Für die Darstellungsformen kann unterschieden werden, ob sich diese auf einzelne Objekte, Objektgruppen oder den gesamten Objektraum auswirken. Der Wirkungsbereich, wie dieser auch in [ROBERTSON 1990, S. 117] als Systematisierungsparameter identifiziert wird, ermöglicht keine eindeutige Einordnung, weil eine Darstellungsmethode für verschiedenen Wirkungsbereiche eingesetzt werden kann.
- *Die menschliche Raumwahrnehmung:* Dieses Merkmal beschreibt die Charakterisierung, die einem Nutzer den Eindruck räumlicher Tiefe auf einer zweidimensionalen Darstellungsfläche vermitteln und umfasst die Informationsquellen, welche der Mensch zur Erkennung des Aufbaus einer dreidimensionalen Szene, insbesondere der Tiefeneigenschaften heranzieht. Eine exakte Einordnung der Darstellungsverfahren wird jedoch durch optische Täuschungen und visuelle Mehrdeutigkeiten erschwert.
- *Das Visualisierungsziel:* Hiermit wird das beabsichtigte Ziel, das unter Einbeziehung einer Visualisierung erreicht werden soll, beschrieben, ohne auf technische Teilhandlungen sowie konkrete Verfahrensschritte einzugehen. Dieses Kriterium entspricht am ehesten einer pragmatischen Herangehensweise.

Darüber hinaus sind weitere Kriterien denkbar, beispielsweise die Echtzeitfähigkeit oder der Automatisierungsgrad eines Abbildungsverfahrens. Für eine – im pragmatischen Sinne – verwendbare Klassifikation sind diese in Bezug auf einen aufgabenorientierten Interfaceentwurf als Systemisierungskriterium ungeeignet. Das Kriterium Visualisierungsziel wird in der Folge für die grundlegende Einteilung gewählt. Zusätzlich zum Klassifikationskriterium ist eine Auswahl an Klassen notwendig, die zur Bewertung von Darstellungstechniken im Hinblick auf ein Visualisierungsziel herangezogen werden. Hierbei dient das *Visual Information Seeking Mantra* nach SHNEIDERMAN (vgl. [SHNEIDERMAN 1996, S. 336–337]), das für eine Vielzahl von Visualisierungssystemen als Referenzmodell dient (vgl. [CRAFT & CAIRNS 2005]), als Ausgangspunkt. Darin beschreibt SHNEIDERMAN, mit dem Grundprinzip *“Overview first, zoom and filter, then details-on-demand”*<sup>52</sup> [SHNEIDERMAN 1996, S. 337], Richtlinien für die Anwendung von Visualisierungen. Ferner wird dadurch ein Rahmenwerk zur Einteilung und Bewertung von Techniken zur Erzeugung von kontextabhängigen Benutzerschnittstellen skizziert. Die Taxonomie von SHNEIDERMAN wird nachfolgend im Kontext der Systematisierung linearer und nichtlinearer Darstellungstechniken erläutert:

- *Der Überblick:* Hierzu zählen Darstellungstechniken, die einen Sichtbereich besitzen, der den gesamten Objektraum in einen Bildraum überführt. Dazu gehören Verfahren, die mittels Bildkomposition oder nichtlinearer Projektion eine umfassende Sicht auf den Objektraum bereitstellen.

<sup>52</sup> „Erstens der Überblick, vergrößern (verkleinern) und filtern, danach Details auf Anforderung“

- *Der Zoom*: Zu dieser Kategorie gehören einerseits Verfahren, die eine *Fokus-und-Kontext-Darstellung* erzeugen. Andererseits werden *Übersicht-und-Detail-Ansichten*, die Einzelheiten und Überblicke in zwei voneinander getrennten Ansichten darstellen, in diese Kategorie eingeordnet.
- *Der Filter*: Diese Gruppe umfasst Darstellungsformen, die eine Gestaltung ausgesuchter Objekte im Bildraum ermöglichen, ohne eine tatsächliche Veränderung des Objektraums herbeizuführen. Es handelt sich dabei um Visualisierungsverfahren, die eine veränderte Sichtbarkeit von Szenenelementen erreichen sowie um kompositorische Darstellungsverfahren zur Anordnung von Objekten im Bildraum.
- *Die Detailinformation*: In diese Kategorie werden Visualisierungsverfahren eingeordnet, die eine Erkennung der Objektform sowie die Bestimmung der Position und Ausrichtung einer visuellen Struktur begünstigen. Dies kann durch multiple Sichten beziehungsweise durch multiprojektive Darstellungsverfahren erreicht werden.
- *Der Vergleich*: Darstellungsformen, durch relationale Aufgaben unterstützt werden können, sind in dieser Kategorie eingeordnet. Dies kann entweder durch die Beibehaltung der Proportion und Ausrichtung von Objekten erreicht werden oder durch die Erhaltung von Flächen oder Winkeln im Bildraum geschehen.
- *Der Verlauf*: In diese Kategorie werden Darstellungsmethoden eingeordnet, welche die Abfolge von Nutzerinteraktionen auf Basis eines beständigen Objektraums veranschaulichen. Dies kann durch multiple Sichten erfolgen oder durch Verfahren, die den Dialogverlauf in die bildstrukturellen Eigenschaften des Bildraums integrieren.

Die Tabelle 2 zeigt, basierend auf dem *Visual Information Seeking Mantra* von SHNEIDERMAN, die Systemisierungskategorien sowie die Umsetzungsmöglichkeiten zum Erreichen der verschiedenen Visualisierungsziele. In den folgenden Unterabschnitten werden die Kategorien zunächst näher erläutert und existierende Darstellungsvarianten vorgestellt. Ferner werden bestehende Abbildungsverfahren, die eine Umsetzung der Visualisierungsziele erreichen, tabellarisch aufgeführt. Dabei werden die Kriterien aufgegriffen, von denen im Zuge der Identifikation des Klassifikationskriteriums Abstand genommen wurde, die aber für eine Anwendungsentwicklung wesentlich sind. Hierbei wird eine Kurzbeschreibung des Verfahrens gegeben und eine Visualisierung, welche aus der angegebenen Quelle stammt und das Resultat des Bilderzeugungsprozesses verdeutlicht, dargeboten. Weiterhin werden die Darstellungsform und die genutzte Modifikationsmöglichkeit der computergrafischen Kamera aufgeführt. Zusätzlich wird über den Wirkungsbereich ausgedrückt, inwieweit sich das Verfahren auf einzelne Elemente des Objektraums, auf Teilbereiche eines Bildraums oder auf das gesamte erzeugte Bild auswirkt. Ferner werden Aussagen zum Interaktivitätsgrad und zum Einsatz des Abbildungsverfahrens in einer interaktiven Anwendung getätigt.

Um die Anwendbarkeit der Systematisierung zu reflektieren und die systematische Verknüpfung der Aufgaben und Ziele eines Nutzers mit den Darstellungsverfahren zu gewährleisten, werden die Evaluationskriterien von BEAUDOUIN-LAFON berücksichtigt (vgl. [BEAUDOUIN-LAFON 2004, S. 17]):

Visualisierungsziel	Umsetzungsmöglichkeit	Bestehende Abbildungsverfahren
Überblick	Vergrößerung des Sichtfeldes	Tabelle 3
	Verdeckungsfreie Darstellung	
Zoom	Übersicht-und-Detail	Tabelle 4
	Fokus-und-Kontext	
Filter	Bildkomposition	Tabelle 5
	Auffaltung	
	Sichtbarkeit	
Detail	Positionseindeutigkeit	Tabelle 6
	Gestaltseindeutigkeit	
	Ausrichtungseindeutigkeit	
Vergleich	Formtreue	Tabelle 7
	Winkel- und Flächentreue	
Verlauf	Comicstrips	Tabelle 8
	Kamerabewegung	

Tabelle 2: Kategorien der Systematisierung von Abbildungsverfahren im Kontext der Visualisierungsziele nach [SHNEIDERMAN 1996, S. 336] im Überblick

- *Beschreibbarkeit:* Die Systematisierung sollte eine möglichst große Menge an Darstellungstechniken beschreiben können.
- *Bewertbarkeit:* Die Systematisierung muss unterschiedliche Darstellungstechniken und Methoden für einen Kontext aufzeigen.
- *Generierbarkeit:* Die Systematisierung muss Designer und Entwickler befähigen, neue Benutzerschnittstellen zu entwerfen.

### 3.5.2 Abbildungsverfahren für Überblicksdarstellungen

Eine überblickshafte Darstellung bietet einem Betrachter eine umfassende Sicht auf einen gegebenen Datenbestand respektive eine visuelle Struktur. Generelle Informationen, Zusammenhänge und Muster innerhalb eines Datenbestandes können oftmals nur in einer Visualisierung erkannt werden, welche die visuelle Struktur gesamtheitlich darstellt. Eine Überblicksdarstellung dient ferner als Ausgangspunkt einer Exploration und ist gleichzeitig wegweisend für die weiterführende Interaktion mit dem System (vgl. [CRAFT & CAIRNS 2005, S. 111]). Eine Modifikation der computergrafischen Kamera bietet einerseits durch die Vergrößerung des Sichtbereiches, andererseits durch die Auflösung von Verdeckungen verschiedene Möglichkeiten eine Überblicksdarstellung zu verwirklichen. Die Tabelle 3 bietet hierzu eine Übersicht über bestehende Visualisierungsverfahren.

Bei zentralprojektiven Darstellungen kann es in Abhängigkeit vom Szenenaufbau zur teilweisen oder vollständigen Verdeckung von Objekten im Bildraum kommen. Die Verdeckung von Elementen im Bildraum ist für den Betrachter ein monoskopischer Tiefenhinweis (vgl. [EYSENCK & KEANE 2010, S. 68 ff.]) und dient bei 3D-Darstellungen auf planaren Ausgabegeräten der Vermittlung eines räumlichen Eindrucks (vgl. [CUTTING 1997, S. 28–29]). Jedoch existiert im Kontext der Überblicksdarstellung das Bestreben nach einer Sichtbarkeit aller Elemente. Eine Auflösung von Verdeckungen für eine umfassende Sicht auf eine Szene kann beispielsweise durch eine Neuordnung der Elemente des Objektraums oder durch eine



Neupositionierung der computergrafischen Kamera erfolgen. In [ELMQVIST 2005] wird ein Verfahren beschrieben, welches unter Beibehaltung von Kameraposition und -ausrichtung durch eine zeitweise Veränderung des Objektraums in Abhängigkeit zur Kamera die Objektverdeckungen minimiert. Durch eine sphärische Expansion werden die Abstände zwischen den Objekten vergrößert und Verdeckungen aufgelöst. Eine weitere Möglichkeit zeigt POPESCU in [POPESCU U. A. 2009] unter Zuhilfenahme eines bildkompositorischen Verfahrens. Der Einsatz mehrerer Kameras mit unterschiedlichen Projektionseigenschaften resultiert in einem bildraumbasiert-multiprojektiven Abbildungsvorgang, durch den eine Bildkomposition auf Basis einer Zusammenführung der Objekte im Bildraum mit minimalen Verdeckungen realisiert wird.

Für eine Überblicksdarstellung ist es notwendig, dass sich die darzustellenden Objekte innerhalb des Kamerasichtkörpers befinden, da dieser den visualisierten Bereich eines Objektraums bei einer computergrafischen Darstellung bestimmt. In Abhängigkeit zur Lage und Ausdehnung der Elemente im Objektraum kann in konventionellen computergrafischen Darstellungen eine große Entfernung zwischen der visuellen Struktur und der Kamera die Folge sein. Die Ursache liegt in der Eigenschaft der zentralprojektiven Abbildung begründet, die einen durch den Sichtkörper definierten Kameraöffnungswinkel von weniger als 180 Winkelgrad festlegt. Die gleichzeitige Abbildung des gesamten Objektraums bedingt folglich eine entfernte Kameraposition. Panoramabilder hingegen zeichnen sich durch einen Sichtbereich von bis zu 360 Winkelgrad aus. Infolgedessen ist die Möglichkeit einer gesamtheitlichen Abbildung des Objektraums gegeben, ohne dass eine explizite Positionierung und Ausrichtung der Kamera außerhalb der visuellen Struktur notwendig ist. Die computergrafische Umsetzung einer Überblicksdarstellung wird in [LÖFFELMANN & GRÖLLER 1996] durch kurvilineare Abbildungen, basierend auf dem Raytracing-Verfahren, erreicht. Durch das verwendete Kameramodell sind krummlinige Darstellungen mit einem variablen Augpunkt möglich, die eine umfassende Sicht auf einen Datenbestand realisieren. Eine weitere Möglichkeit besteht in der Komposition von Einzelabbildern als multiprojektiver Abbildungsvorgang. Hierbei werden, wie durch RADEMACHER und BISHOP unter Zuhilfenahme einer Kamerafahrt aufgezeigt (vgl. [RADEMACHER & BISHOP 1998]), Einzelabbilder zu einer Überblicksdarstellung zusammengefügt, um die Darstellung aller Elemente eines Objektraums zu erreichen.

### 3.5.3 Abbildungsverfahren für den Zoom

Die Visualisierung großer Datenmengen ist durch die Begrenzungen des Ausgabegerätes und der grafischen Auflösung limitiert. Diesen Beschränkungen wird entweder durch den Verzicht auf Detailinformationen oder durch die Reduktion des Sichtbereiches zugunsten der Visualisierung ausgewählter Elemente des Objektraums entgegengewirkt. Zusätzlich zu den technischen Beschränkungen sind die kognitiven Fähigkeiten des Betrachters bei der visuellen Erfassung und Verarbeitung von großen Datenmengen ebenfalls begrenzt (vgl. [PREIM & DACHSELT 2010, KAP. 2; SCHUMANN & MÜLLER 2000, S. 107 FF.]). Ausgehend von einer Überblicksdarstellung ist der Prozess des Zoomens ein Vorgang zur Reduzierung der Datenmenge in einer Darstellung, indem durch eine geometrische Veränderung des Sichtbereiches die Menge an Daten im Bildraum gezielt reduziert wird. Das Hineinzoomen entfernt situativ.

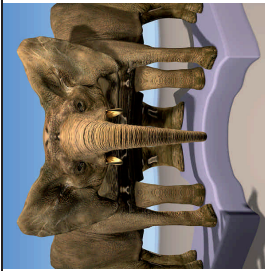
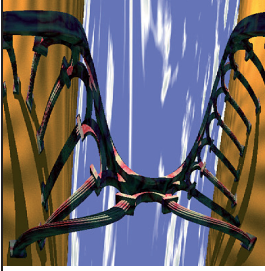
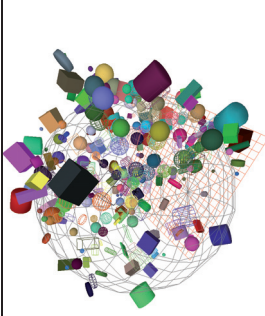
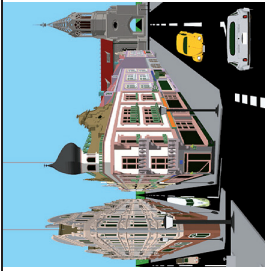
Überblick	Vergrößerung des Sichtbereiches		Verdeckungsfreie Darstellung	
Kurzbeschreibung	Überblicksansicht durch Komposition von Einzelabbildern, die durch eine Kamerafahrt um ein Objekt aufgezeichnet werden.	Der Einsatz von geometrischen Figuren als Projektionsstrahlenermittler erlaubt die Erzeugung nichtlinearer Visualisierungen von Szenenübersichten.	Der Einsatz einer kameraabhängigen, sphärischen Expansion des Objektraums dient der Darstellung aller Elemente in einer Szene.	Durch die komponierte Darstellung von Kamerabildern mit unterschiedlichen Augpunkten und Ausrichtungen wird eine Szenenübersicht erzeugt.
Abbildung				
Literaturquelle	[RADEMACHER & BISHOP 1998]	[LÖFFELMANN & GRÖLLER 1996]	[ELMQVIST 2005]	[POPESCU U. A. 2009]
Automatisierungsgrad	Automatischer Abbildungsvorgang nach Festlegung des Kamerapfades	Die Abbilderzeugung wird nach der Kameradefinition initiiert	Interaktive Anpassung des Bildraums durch Nutzerinteraktion	Manuelle Festlegung der Kameras mit automatischer Bildkomposition
Umsetzung in interaktiver Echtzeitanwendung	Nicht gegeben	Nicht gegeben	Gegeben	Gegeben
Wirkungsbereich	Global	Global	Lokal	Global
Darstellungsform	Bildraumbasierte Multiperspektive	Krummlinige Perspektive	Objektraumbasierte Multiperspektive	Bildraumbasierte Multiperspektive
Manipulationsgegenstand	Bildebene	Projektionsstrahlen	Objekt	Bildebene

Tabelle 3: Übersicht über die Abbildungsverfahren für Überblicksdarstellungen

unwichtige Informationen, um gleichzeitig die wichtigen Elemente zu fokussieren und in der reduzierten Darstellung zu organisieren. Indes ermöglicht das Hinauszoomen das Aufdecken von Informationen, die womöglich bereits bekannt, aber aus dem Fokus des Anwenders gelangt sind. Darüber hinaus ist durch ein Hinauszoomen eine Wiederentdeckung der Position innerhalb des Objektraums möglich, wodurch die aufgedeckten Detailinformationen in einen Kontext gesetzt werden können. In Tabelle 4 werden wichtige Verfahren für die Umsetzung eines bildbasierten Zooms aufgezeigt und gegenübergestellt.

Da der Vorgang des Zoomens durch die Bewegung der Kamera auf eine Position im Objektraum oftmals nicht ausreichend für den Interaktionsprozess ist oder einen Verlust von Bezugspunkten und Verhältnissen in einem zentralprojektiven Abbild herbeiführt (vgl. [LAMAR U. A. 2001; WANG U. A. 2005]), wird der Prozess durch nichtlineare Darstellungstechniken unterstützt. Die Veränderung des Bildraums durch einen Zoomvorgang wird in computergrafischen Anwendungen einerseits durch Übersicht-und-Detail-Visualisierungen, andererseits durch Fokus-und-Kontext-Darstellungen unterstützt. Fokus-und-Kontext-Visualisierungen bieten eine Darstellungsform, in der einzelne Objekte oder Bildbereiche aus dem Gesamtkontext herausgehoben werden. In [CARPENDALE U. A. 1996] wird eine solche Herausstellung durch die geometrische Vergrößerung einzelner Objekte herbeigeführt, wobei diese im strukturellen Aufbau der Szene verankert bleiben. Weiterhin zeigen LORENZ U.A. sowie PASEWALDT U. A. anhand von virtuellen Stadtmodellen, dass durch eine Deformation der gesamten visuellen Struktur eine Fokussierung einzelner Bereich unter Beibehaltung des Szenenkontextes realisiert werden kann (vgl. [LORENZ U. A. 2008; PASEWALDT U. A. 2011]). Ferner wird durch eine Modifikation der Projektionsstrahlen auf Basis des Raytracing-Verfahrens in [VALLANCE & CALDER 2001B] die Hervorhebung von Szenenbereichen im Bildraum erreicht. Im letztgenannten Verfahren erfolgt dabei der Zoom unabhängig von den konkreten Elementen einer Szenerie beziehungsweise einer Veränderung der visuellen Struktur. Ein alternativer Ansatz ist durch Übersicht-und-Detail-Darstellungen gegeben. Durch den Einsatz von zwei oder mehr simultan dargebotenen Abbildern werden eine Übersichts- und eine Detailansicht getrennt voneinander dargeboten. Eine Verknüpfung der Visualisierungen erfolgt durch eine Kopplung von Darstellungsparametern, beispielsweise die Ausrichtung der computergrafischen Kamera, wodurch eine semantische Verknüpfung entsteht. Bestehende Verfahren zeigen einerseits die Umsetzung dieses Ansatzes durch umschließende Ansichten wie es in [STOAKLEY U. A. 1995] für immersive Umgebungen konzipiert und umgesetzt ist. Dies bietet dem Anwender die Möglichkeit, die eigene Position innerhalb der Szenenstruktur zu bestimmen und gleichzeitig die Darstellung ausgewählter Szenenbereiche zu erfassen. Andererseits wird in [PLUMLEE & WARE 2003A] die Benutzerschnittstelle in zwei nebeneinander angeordnete Sichten unterteilt, um einen fokussierten Ausschnitt einer visuellen Struktur in Relation zur Gesamtdatenmenge zu veranschaulichen.

### 3.5.4 Abbildungsverfahren für das Filtern

Während das Zoomen eine geometrische Anpassung des Bildraums beschreibt, entspricht das Filtern einer Relevanzbestimmung der Datenmenge. Im Kontext der computergrafischen Projektion bietet das Filtern die Möglichkeit, einen Bildraum zu erzeugen, der selektierte

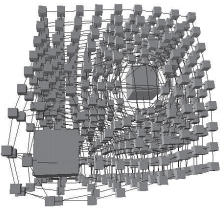
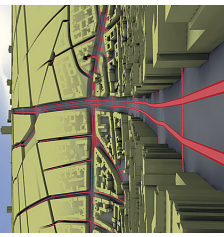
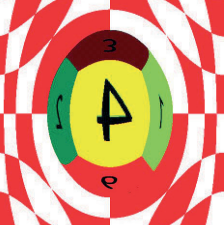
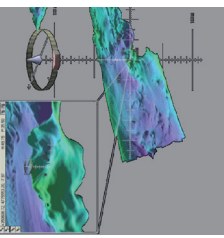
Zoom	Fokus-und-Kontext		Übersicht-und-Detail	
Kurzbeschreibung	Eine vergrößerte Darstellung einzelner Objekte durch Skalierung wobei umliegende Objekte verschoben werden.	Eine Deformation der gesamten visuellen Struktur bildet die Grundlage für die Fokussierung eines Bereiches im Gesamtkontext.	Gewölbte Flächen als Emitter der Projektionsstrahlen erlauben die Vergrößerung von Bereichen im Bildraum.	Eine Miniaturansicht der gesamten Szene wird in die Visualisierung eingebettet und nutzerspezifisch ausgerichtet.
Abbildung				
Literaturquelle	[CARPENDALE U. A. 1996]	[PASEWALDT U. A. 2011]	[VALLANCE & CALDER 2001B]	[PLUMLEE & WARE 2003A]
Automatisierungsgrad	Automatische Anpassung des Objektraums infolge einer Nutzerinteraktion	Interaktive Änderung der Darstellungsparameter und Anpassung zur Laufzeit	Automatischer Abbildungsvorgang nach Definition der Emitterfläche	Wechselseitige, automatische Anpassung der Sichten bei der Nutzerinteraktion
Umsetzung in interaktiver Echtzeitanwendung	Gegeben	Gegeben	Gegeben	Gegeben
Wirkungsbereich	Objektspezifisch	Global	Lokal	Global
Darstellungsform	Objektraumbasierte Multiperspektive	Objektraumbasierte Multiperspektive	Krummlinige Perspektive	Multiple Sicht
Manipulationsgegenstand	Objekt	Objekt	Projektionsstrahlen	Zentrum der Projektion

Tabelle 4: Übersicht über die Abbildungsverfahren für den Zoom

Elemente darbietet. Ferner können Abbildungsverfahren eingesetzt werden, die eine gewichtete Ordnung der Objekte im Bildraum realisieren und in der Folge weniger relevante Elemente in periphere Bildraumbereiche verlagern, ohne diese vollständig aus dem Bildraum zu entfernen. Durch diese Neuordnung der Objekte im Bildraum ist es möglich, die nutzerdefinierte Wertigkeit der Daten zu verdeutlichen. Die wesentlichen Abbildungsverfahren zur Anordnung und Sichtbarkeit visueller Strukturen werden in der Tabelle 5 in Kurzform dargestellt.

Im zentralprojektiven Bildraum ist die Anordnung der Elemente in Abhängigkeit zur geometrischen Struktur im Objektraum sowie der Position und Ausrichtung der computergrafischen Kamera festgelegt. Außer einer Neupositionierung der Elemente im Objektraum kann ferner durch die Modifikation des Abbildungsprozesses die Anordnung der Objekte im Bildraum zielgerichtet beeinflusst werden. Mittels einer Bildkomposition zeigt AGRAWALA, dass separat gerenderte Objekte im Bildraum entsprechend der Nutzervorgaben platziert werden können (vgl. [AGRAWALA U. A. 2000]). Durch eine Manipulation der Projektionsstrahlen wird in [SINGH 2002] die Sichtbarkeitsreihenfolge objektspezifisch festgelegt. Dadurch wird es möglich, dem Anwender eine Darstellung des Objektraums bereitzustellen, die eine verbesserte Sichtbarkeit von nutzerrelevanten Informationen und Zusammenhängen bietet.

Neben einer Strukturierung der Elemente im Bildraum mithilfe eines kompositorischen Ansatzes kann durch eine veränderte Sichtbarkeit von Elementen und Bereichen eine Filterung erfolgen. HSU U. A. nutzen in [HSU U. A. 2011] ein multiprojektives Abbildungsverfahren, um verdeckte – jedoch situativ relevante – Objekte mit separaten Kameras abzubilden. Durch eine Bildkomposition bestehend aus Einzelabbildern werden die expliziten Bereiche der visuellen Struktur zusammengeführt. Ein vergleichbarer Ansatz wird in [SUDARSANAM U. A. 2008] beschrieben, um relevante Szenenbereiche für den Betrachter sichtbar zu machen. SUDARSANAM U. A. zeigen, dass durch eine objektspezifische Definition von Sichtkörpern zusätzlicher Kameras situativ irrelevante Szenenbereiche aus dem Abbildungsvorgang herausgefiltert werden können. Hierdurch können Szenenbereiche in den Bildraum überführt werden, die durch eine Standardabbildung nicht auf die Bildebene projiziert werden, da diese beispielsweise durch ineinander verschachtelten visuellen Strukturen verdeckt sind.

### 3.5.5 Abbildungsverfahren für Detaildarstellungen

In interaktiven Anwendungen kann eine große Anzahl von Informationen in einer Übersichtsdarstellung verbildlicht werden. In Abhängigkeit von der Datenmenge ist es jedoch oftmals die Limitationen der Darstellungsfläche, die eine Visualisierung der Daten, in einem für den Nutzer und den Kontext notwendigen Detailgrad, nicht zulässt. Eine fokussierte Darstellungsweise ist notwendig, um beispielsweise spezifische Informationen zu einem Element innerhalb einer großen Datenmenge zu erhalten oder dessen Attribute detailliert zu veranschaulichen. Eine Zusammenfassung von Verfahren, die eine Visualisierung von Detailinformationen in 3D-Anwendungen begünstigen, erfolgt in Tabelle 6.

In 3D-Darstellungen können jedoch auch bei einer detaillierten und fokussierten Darstellungsweise Informationen durch den zentralprojektiven Abbildungsvorgang mehrdeutig


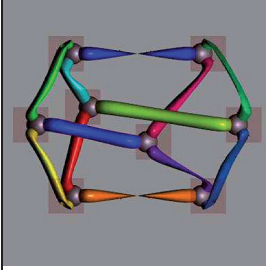
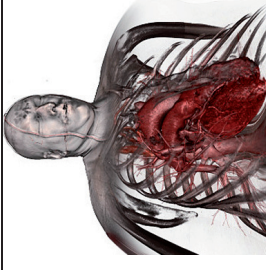
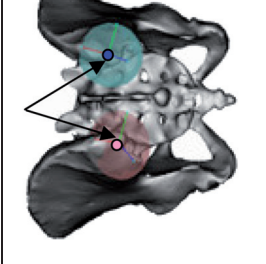
Filter	Anordnung		Sichtbarkeit	
Kurzbeschreibung	Freie Positionierung und Ausrichtung von Einzelkameras erlaubt eine individuelle, manuelle Komposition von Elementen im Bildraum.	Erzeugung eines multiperspektivischen Bildes auf Basis von nutzerspezifisch gewichteten Einzelkameras, zur Festlegung der Sichtbarkeitsreihenfolge.	Einzelkameras erzeugen Abbilder in unterschiedlichen Detailbereichen, die auf Basis von Maskierungsfunktionen in ein Bild zusammengeführt werden.	Einzelkameras mit verschiedenen Sichtkörperdefinitionen ermöglichen in der Zusammenführung in ein Bild, das Herausfiltern von Objekten oder Objektbestandteilen.
Abbildung				
Literaturquelle	[AGRAWALA U. A. 2000]	[SINGH 2002]	[HSU U. A. 2011]	[SUDARSANAM U. A. 2008]
Automatisierungsgrad	Manuelle Festlegung der Kameraparameter und Sichtbarkeiten vor der Abbilderzeugung	Manuelle Festlegung der Kameraposition und Gewichtung und anschließender Bilderzeugung	Interaktive Anpassung der Kameraeigenschaften und automatische Bildsynthese	Interaktive Festlegung der Kamerasishtkörper mit automatischer Bilderzeugung
Umsetzung in interaktiver Echtzeitanwendung	Nicht gegeben	Nicht gegeben	Gegeben	Gegeben
Wirkungsbereich	Objektspezifisch	Lokal	Lokal	Objektspezifisch
Darstellungsform	Objektraumbasierte Multiperspektive	Bildraumbasierte Multiperspektive	Bildraumbasierte Multiperspektive	Objektraumbasierte Multiperspektive
Manipulationsgegenstand	Bildebene	Bildebene	Bildebene und Projektionsstrahlen	Bildebene und Sichtkörper

Tabelle 5: Übersicht über die Abbildungsverfahren für das Filtern



dargestellt werden. Es entstehen visuelle Uneindeutigkeiten bezüglich der Position und der Ausrichtung von Objekten, die über alle drei Raumachsen hinweg gegeben sind. Die Ursache dessen sind die fehlenden Tiefenhinweise einer räumlichen Szene auf der planaren Darstellungsfläche. Eine Auflösung der Mehrdeutigkeit kann durch die Integration zusätzlicher Informationen erreicht werden, die zu einer Verbesserung der Raumwahrnehmung des Betrachters führen (vgl. [CUTTING 1997, S. 28–29]). Beispielsweise kann der Schattenwurf eines Objektes oder die Sichtbarkeit eines Horizontes eine Mehrdeutigkeit verringern oder aufheben. Darüber hinaus kann durch die Modifikation der computergrafischen Kamera eine Eindeutigkeit im Hinblick auf die Position und die Ausrichtung von Objekten erreichen werden. In [MAPLE U. A. 2004] werden zwei Sichten miteinander gekoppelt, um die Positions- und Ausrichtungsbestimmung des Nutzers innerhalb des Objektraums und in Bezug auf die visuellen Strukturen zu verbessern. Dies ermöglicht eine exakte Positionsbestimmung des Anwenders innerhalb des Objektraums. Überdies kann durch die Anpassung der bildstrukturellen Eigenschaften die Positionsbestimmung von Objekten in virtuellen Szenen unterstützt werden. In [COLEMAN U. A. 2005] werden für eine Bildraummanipulation grafische Werkzeuge eingesetzt, die jeweils Kameras mit spezifischen Parametern symbolisieren. Basierend auf der Platzierung der Kameras werden die Teilbilder erzeugt und als multiprojektive Visualisierung dargeboten. Hierdurch können bildstrukturelle Eigenschaften definiert werden, durch die eine Positionsbestimmung von Objekten verbessert wird.

Neben der Auflösung von Mehrdeutigkeiten hinsichtlich der Betrachter- und der Objektposition ist die Bestimmung von Objekteigenschaften, beispielsweise die Gestalt, eine Voraussetzung für die Erfassung von Detailinformationen. Das Erkennen der sichtbaren Form von Objekten wird in [MASHIO U. A. 2010] durch exklusive Sichten auf einzelne Bereiche einer Szene unterstützt, wodurch ferner die Reduktion perspektivischer Verzerrungen erreicht wird. Das Verfahren ermöglicht die Beibehaltung der Objektgestalt in virtuellen Szenen durch die Festlegung von individuellen Kameraaugpunkten und der nachfolgenden Zusammenführung der zentralprojektiven Teilbilder. In [TORY U. A. 2004] werden hingegen zusätzliche Bildebenen mit orthografischen Ansichten in den Bildraum eingefügt, die um das Objekt oder die Szene positioniert werden. Dadurch wird eine Auflösung der Ausrichtungsmehrdeutigkeit erreicht [TORY U. A. 2004, S. 77–78]. Ferner wird in [ZAVESKY 2011] am Beispiel anthropomorpher Formen aufgezeigt, dass durch eine Veränderung der Objektgeometrie in Abhängigkeit zur Kamera die Eindeutigkeit der Objektausrichtung herbeigeführt werden kann.

### 3.5.6 Abbildungsverfahren für den Vergleich

Außer den objektbezogenen Operationen bilden ebenso relationale Aufgaben, beispielsweise der Vergleich von Elementen oder die Ermittlung von Korrelationen, wesentliche Ziele des Nutzers im interaktiven Prozess mit einem interaktiven System. Der computergrafische Abbildungsprozess bietet Möglichkeiten, die Durchführung relationaler Aufgaben zu unterstützen, wodurch der Kommunikationsprozess zwischen Nutzer und System durch die Gestaltung des Abbildes begünstigt wird. Wesentliche Verfahren zur Verwirklichung von Visualisierungsformen für den Vergleich werden in Tabelle 7 aufgezeigt.



Detail	Position	Gestalt	Ausrichtung
Kurzbeschreibung	Die Integration einer zusätzlichen, gekoppelten Sicht dient in virtuellen Umgebungen als Orientierungshilfe für den Nutzer.	Mithilfe von Widgets, die Kameras mit fixen Parametern beschreiben, werden Einzelbilder erzeugt und nutzerabhängig ineinander geblendet.	Zusätzliche orthografische Sichten werden um Objekte positioniert und erlauben die Identifikation der Objektausrichtung in zentralprojektiven Darstellungen.
Abbildung			
Literaturquelle	[MAPLE U. A. 2004]	[COLEMAN U. A. 2005]	[ZAVESKY 2011]
Automatisierungsgrad	Automatische Anpassung der gekoppelten Sicht bei der Nutzerinteraktionen	Automatische Erzeugung des Bildraums auf Basis von Kamerafestlegungen	Objektauswahl mit anschließender automatischer Bilderzeugung
Umsetzung in interaktiver Echtzeitanwendung	Gegeben	Gegeben	Gegeben
Wirkungsbereich	Global	Lokal	Objektspezifisch
Darstellungsform	Multiple Sicht	Bildraumbasierte Multiperspektive	Objektraumbasierte Multiperspektive
Manipulationsgegenstand	Zentrum der Projektion	Bildebene	Objekt

Tabelle 6: Übersicht über die Abbildungsverfahren für Detaildarstellungen

Ein Verfahren, das in [FRANKE U. A. 2007] vorgestellt wird, ermöglicht durch eine geometrische Veränderung der Objekte eine Abbildung der visuellen Strukturen, unter Beibehaltung der Objektproportionen unabhängig von deren positioneller Verortung im Bildraum. Weiterführend wird in [ZAVESKY U. A. 2010] gezeigt, dass durch eine individuelle Anpassung der Objektausrichtung die Vergleichbarkeit von Objekten sichergestellt werden kann. AGRAWALA veranschaulicht in [AGRAWALA U. A. 2000] durch ein konzeptionell vergleichbares Vorgehen, dass durch den Einsatz einer diskreten multiprojektiven Darstellungen eine Proportionstreuung von Objekten unter Beibehaltung der ursprünglichen Objektposition erreicht wird. Eine weitere Möglichkeit relationale Aufgaben in einer 3D-Benutzerschnittstelle durch eine Modifikation der computergrafischen Bildsynthese zu unterstützen, wird durch eine Darstellung von visuellen Strukturen in koordinierten multiplen Sichten erreicht. In [ROBERTS 2007, S. 64] wird diese spezielle Form der koordinierten Sichten als *Master-Slave-Verhältnis* beschrieben. Diese Verknüpfung von zwei Sichten wird erreicht, indem eine Sicht in eine definierte Abhängigkeit zu einer zweiten Sicht gesetzt wird. STARKE U. A. zeigen in [STARKE U. A. 2011], dass diese Form der koordinierten Sichten genutzt werden kann, um vergleichende Aufgaben im Bildraum durchzuführen. Die für eine Vergleichbarkeit notwendige Formtreue wird dabei durch die zentrierte Positionierung der Objekte im Bildraum sichergestellt. Nicht immer ist jedoch die Gestalt eines Objektes das gewünschte Vergleichsattribut des Anwenders. GLAESER und GRÖLLER untersuchen hierzu in [GLAESER & GRÖLLER 1999] verschiedene Verfahren der Kartenprojektion für den Einsatz in virtuellen Umgebungen. Die Autoren zeigen, dass unterschiedliche kurvilineare Abbildungsfunktionen verschiedene Eigenschaften von Objekten und Szenen, beispielsweise *Winkel-* oder *Flächentreue*, in den Bildraum überführen und dadurch eine visuelle Vergleichbarkeit von speziellen Objekteigenschaften begünstigen. Trotz einer Auflösung der Gestalteigenschaften eines Elementes können somit Distanz- und Volumenvergleiche zwischen Objekten erreicht werden.

### 3.5.7 Abbildungsverfahren für Verlaufsdarstellungen

Der interaktive Prozess im Nutzer-Bild-Dialog besteht nur in speziellen bildgestützten Interaktionen aus einem einzigen Bildsyntheseschritt, um ein Visualisierungsziel zu erreichen. Überwiegend ist es ein Vorgang, der durch mehrere zielgerichtete Veränderungen des Bildraums gekennzeichnet ist. Die Darstellung des Interaktionsverlaufes und die Möglichkeit zu einem früheren Zustand innerhalb des Prozesses zurückzukehren, sind daher wesentliche Bestandteile im interaktiven Prozess mit einem interaktiven System. Durch die Nachvollziehbarkeit des Verlaufes ist es möglich, den aktuellen Zustand der Darstellung mit einem früheren Zustand in Bezug zu setzen. Außerdem bietet es die Möglichkeit Interaktionen rückgängig zu machen, um daraufhin den Interaktionsprozess in einer veränderten Form fortzuführen. Durch die Möglichkeit eines abgeänderten Fortgangs der Interaktion kann der Nutzer daraus resultierende Informationen nutzen und kombinieren. Visualisierungstechniken können die Aktivitäten eines Nutzers diesbezüglich unterstützen, indem die Darstellungen einen Verlauf vollzogener Aktivitäten darbieten. Dies kann einerseits durch multiple Sichten, andererseits mithilfe nichtlinearer Darstellung verwirklicht werden. Die Verfahren zur Verlaufsdarstellung, die nachfolgend vorgestellt werden, sind in Tabelle 8 in kompakter Form gegenübergestellt.

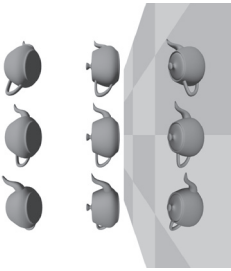
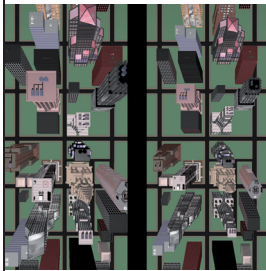
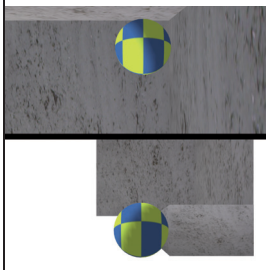

Vergleich	Form	Winkel und Flächen			
Kurzbeschreibung	Eine kontinuierliche Veränderung der Objektgeometrie in Abhängigkeit zur Kameraposition und -ausrichtung ermöglicht eine Vergleichbarkeit der Objektgestalt.	Das Abbildungsverfahren für eine Vergleichbarkeit von Objekten wird durch eine objektraumbasierte multiprojektive Darstellung erreicht.	Koordinierte multiple Sichten auf Basis von Kamerakopplungen mit separaten Projektionsvorgängen ermöglichen einen visuellen Vergleich von Objekten.	Kurvilineare Projektionen bieten eine Überführung von winkel- oder flächentreuen Darstellungen der Objekte in den Bildraum.	
Abbildung					
Literaturquelle	[FRANKE U. A. 2007]	[AGRAWALA U. A. 2000]	[STARKE U. A. 2011]	[GLAESER & GRÖLLER 1999]	
Automatisierungsgrad	Objektauswahl und anschließende automatische Abbilderzeugung	Festlegung der Kameraparameter mit anschließender Abbilderzeugung	Festlegung der Kopplungsparameter mit anschließender Abbilderzeugung	Festlegung des Projektionsverfahrens mit anschließender automatischer Bildsynthese	
Umsetzung in interaktiver Echtzeitanwendung	Nicht gegeben	Nicht gegeben	Gegeben	Nicht gegeben	
Wirkungsbereich	Objektspezifisch	Objektspezifisch	Global	Global	
Darstellungsform	Objektraumbasierte Multiperspektive	Objektraumbasierte Multiperspektive	Multiple Sicht	Krummlinige Perspektive	
Manipulationsgegenstand	Objekt	Bildebene	Zentrum der Projektion	Projektionsstrahlen	

Tabelle 7: Übersicht über die Abbildungsverfahren für den Vergleich

In [SHAMIR U. A. 2006] dienen multiple Sichten der Vereinfachung und Zusammenfassung von Navigations- und Interaktionsverläufen des Nutzers in einer 3D-Umgebung, indem essenzielle Schritte im Nutzer-Bild-Dialog als Comicstrip dargestellt werden. Hierdurch wird eine kompakte Darstellung der wesentlichen Sichten auf den Objektraum festgehalten, die dem Anwender bereits dargeboten wurden. In [KEEFE U. A. 2009] wird der Ansatz multipler Sichten für die Veranschaulichung einer Datenveränderung verwendet. Je nachdem wie die Sichten auf eine visuelle Struktur gewählt sind, können unterschiedliche Abschnitte des Vorganges der Datenveränderung im zeitlichen Verlauf in multiplen Sichten exploriert werden.

Ein alternativer Ansatz zu mehreren simultan dargebotenen Sichten, die den Verlauf in einer Sequenz von Einzelabbildern aufzeigen, kann dieser in einem einzigen Bild dargeboten werden. Das Konzept zeichnet sich dadurch aus, dass der Interaktionsverlauf in die bildstrukturellen Eigenschaften des Bildraums integriert ist. In [SINGH & BALAKRISHNAN 2004] werden die Bewegungen der Kamera während der Interaktion des Nutzers aufgezeichnet. Parallel dazu erfolgt auf Basis vergangener und aktueller Kameraeinstellungen der Aufbau des Bildraums. Bereiche, die bereits exploriert wurden, werden in einer vergrößerten Darstellung im Bildraum verortet, um den Interaktionsverlauf aufzuzeigen. Für die Herbeiführung der Änderungen verwenden SINGH und BALAKRISHNAN ein Transformationsgitter. Dieses dreidimensionale Gitter wird auf bereits durch den Nutzer fokussierte Bereiche gelegt und deformiert diesen Teil des Objektraums in Abhängigkeit der Kameraparameter. Die Autoren zeigen ferner eine alternative Verfahrensweise, die ebenso der Verdeutlichung des Interaktionsverlaufes des Nutzers dient. Hierbei werden Einzelkameras genutzt, deren Fokusse auf die Bereiche des Objektraums gelegt sind, die im Interaktionsverlauf vom Anwender bereits betrachtet wurden. Eine weitere Kamera rendert die Szene mit lokalen Vergrößerungen an den Fokuspunkten der Kameras, wodurch das Durchwandern und Verweilen des Nutzers innerhalb eines Objektraums im Bildraum abgebildet wird. Ein weiterer Ansatz, der die Darstellung eines Interaktionsverlaufes verbildlicht, ist *Pushbroom*. Mithilfe des Pushbroom-Verfahrens können multiperspektivische Darstellungen erzeugt werden, welche die zeitlichen Unterschiede zwischen verschiedenen Perspektiven aufzeigen. Realisiert wird dies durch eine zeitliche Aneinanderreihung von Abbildern. Dabei werden einzelne Kamerabilder in einem *spatiotemporalen Volumen* (vgl. [SEITZ & KIM 2003, S. 17]) zusammengefügt. Das Volumen dient der Erzeugung einer Pushbroom-Darstellung, indem dieses nicht in der zeitlichen Abfolge verbildlicht wird, sondern die Einzelbildebene orthogonal zur ursprünglichen Bildfolge in den Abbildungsvorgang eingebunden werden (vgl. [ROMAN U. A. 2004, S. 541–542]). Durch diese Form multiprojektiver Darstellung werden Veränderungen des Objektraums und der Kameraparameter in den Bildraum überführt. Das resultierende multiperspektivische Bild zeigt in der Folge Veränderungen über die Zeit.

## 3.6 Diskussion und Motivation

Nach einer Einführung in die Thematik der 3D-Benutzerschnittstellen wird im Unterabschnitt 3.1.5 anhand wissenschaftlicher Arbeiten die Relevanz einer Systematisierung von Visualisierungsformen verdeutlicht. Ferner werden in Abschnitt 3.2 Darstellungsweisen erläutert, die

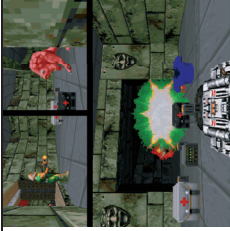
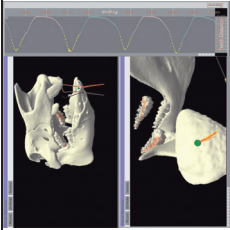
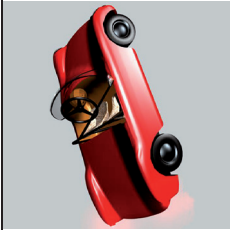
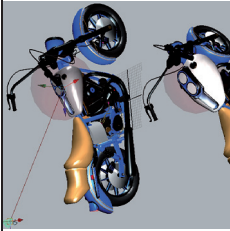

Verlauf	Einzelansichten		Kamerabewegung		
Kurzbeschreibung	Der Interaktionsverlauf des Anwenders in einer virtuellen Umgebung wird in chronologisch geordneten Einzelbildern dargestellt.	Multiple Sichten erlauben die Visualisierung räumlicher und zeitlicher Veränderungen in einer simultanen Darstellung.	Betrachtete Bildbereiche werden vergrößert dargestellt und geben den Interaktionsverlauf als multiprojektives Bild wieder.	Objekte einer Szene werden entsprechend der Interaktion und Aufmerksamkeit des Nutzers geometrisch verändert.	Die Abbildung eines spatiotemporalen Volumens erlaubt die Visualisierung von Interaktionsverläufen in einem einzelnen Bild.
Abbildung					
Literaturquelle	[SHAMIR U. A. 2006]	[KEEFE U. A. 2009]	[SINGH & BALAKRISHNAN 2004]	[SINGH & BALAKRISHNAN 2004]	[SEITZ & KIM 2003]
Automatisierungsgrad	Automatische Bilderzeugung und manuelle Erstellung des Comicstrips	Automatische Erzeugung koordinierter Sichten auf Basis von Nutzervorgaben	Festlegung der Kameraparameter mit anschließender Bildsynthese	Festlegung eines Deformationsgitters mit anschließender Bildsynthese	Manuelle Festlegung des Kamerapfades zur Erzeugung des spatiotemporalen Volumens
Umsetzung in interaktiver Echtzeitanwendung	Nicht gegeben	Gegeben	Nicht gegeben	Nicht gegeben	Nicht gegeben
Wirkungsbereich	Global	Global	Lokal	Lokal	Global
Darstellungsform	Multiple Sicht	Multiple Sicht	Bildraumbasierte Multiperspektive	Objektraumbasierte Multiperspektive	Bildraumbasierte Multiperspektive
Manipulationsgegenstand	Zentrum der Projektion	Zentrum der Projektion	Bildebene	Objekt	Bildebene

Tabelle 8: Übersicht über die Abbildungsverfahren für Verlaufsdarstellungen

abseits der Ähnlichkeitsparadigmen und des computergrafischen Naturalismus entstanden und eine gezielte Kommunikation von Bildaussagen auf Basis der zugrunde liegenden bildstrukturellen Eigenschaften aufzeigen. Im Anschluss an diese Erläuterungen folgen im Abschnitt 3.3 Betrachtungen zur computergrafischen Bildsynthese sowie dem Kameramodell. Dabei werden auf Basis des Kameramodells Möglichkeiten identifiziert, den Prozess der Abbilderzeugung zu modifizieren. Die Analyse der Manipulationsmöglichkeiten bildet die Grundlage für eine Systematisierung von Abbildungsverfahren auf Basis der projektionsbedingten bildstrukturellen Eigenschaften. Die Einordnung der Abbildungsverfahren zeigt eine begründete Unterteilung in lineare, nichtlineare und multiprojektive Abbildungsverfahren. Aufbauend auf dieser verfahrenstechnischen Betrachtung werden in Abschnitt 3.4 Gestaltungshinweise gegeben, die im Abschnitt 3.5 in eine Systematisierung von Darstellungsformen im Hinblick auf eine aufgaben- und kontextbezogene Gestaltung von 3D-Benutzerschnittstelle interaktiver Anwendungen einfließen. Aus einer umfangreichen Recherche verwandter Arbeiten werden Einsatzmöglichkeiten von Darstellungsformen identifiziert und klassifiziert, wobei wichtige Vertreter tabellarisch aufgeführt sind. Dabei werden mannigfaltige Darstellungstechniken vorgestellt und deren Anwendung im jeweiligen konkreten Anwendungskontext aufgezeigt.

Es ist deutlich geworden, dass lineare, nichtlineare und multiprojektive Verfahren eingesetzt werden können, um den Bildraum von 3D-Benutzerschnittstellen gezielt zu gestalten. Mithilfe der aufgezeigten Verfahren wird ein Repertoire an Darstellungsverfahren umrissen, durch das aufgaben- und kontextbezogene Ziele eines Anwenders unterstützt werden können. Dies umfasst Darstellungsweisen, welche die Grundprinzipien des *Visual Information Seeking Mantra* in Form vielfältiger Visualisierungstechniken konkretisieren und auf interaktive 3D-Benutzerschnittstellen übertragen. Einzig der von SHNEIDERMAN identifizierte Aspekt der Extraktion (vgl. [SHNEIDERMAN 1996]), der beinhaltet, dass eine Benutzerschnittstelle aus einer Anwendung in eine andere überführbar sein sollte, ist in der gezeigten Systematisierung nicht thematisiert. Dieser vielmehr softwaretechnologische Aspekt ist nicht durch eine Manipulation der computergrafischen Kamera zu realisieren und wurde infolgedessen nicht als Systematisierungsaspekt berücksichtigt.

Aus den Erkenntnissen dieses Kapitels ergeben sich zwei Anforderungen für einen leistungsfähigen und gleichzeitig nutzerzentrierten Einsatz von Visualisierungstechniken in interaktiven 3D-Anwendungen: Einerseits sollten computergrafische Abbildungsverfahren für eine einfache und flexible Nutzbarkeit und Wiederverwendbarkeit in interaktiven Anwendungen bereitgestellt werden. Andererseits soll, unter Einbeziehung dieser Verfahren, der Entwurfsprozess von Benutzerschnittstellen durch effiziente Modelle und effektive Werkzeuge für die Entwicklung von interaktiven 3D-Anwendungen unterstützt werden.



## 4 BiLL: Ein Framework für den 3D-Interfaceentwurf

Eine reichhaltige Palette von linearen und nichtlinearen Visualisierungsverfahren steht in einer Vielzahl von proprietären 3D-Anwendungen zur Verfügung. Einerseits sind die Verfahren in Demonstratoranwendungen integriert, um deren Machbarkeit aufzuzeigen. Andererseits werden die Visualisierungstechniken zur Lösung spezifischer Herausforderungen eingesetzt und infolgedessen auf die Anforderungen und Rahmenbedingungen im Problemfeld der Anwendung angepasst. Die dadurch entstehenden Insellösungen haben die Beschränkung, dass ein flexibler Einsatz der Methoden und ferner deren kompositorische Nutzung nicht vorgesehen beziehungsweise möglich sind. Demgegenüber ist für die Integration sowie den Einsatz von verschiedenen Darstellungstechniken in einem modularen Visualisierungssystem eine wesentliche Vorarbeit notwendig sowie ein vergleichsweise hoher Einarbeitungsaufwand erforderlich. Dieser hohe Aufwand ermöglicht es jedoch, auch kombinatorische Visualisierungen zu entwerfen. Um den Herausforderungen zur Einbindung von vielfältigen Darstellungsweisen in 3D-Benutzerschnittstellen zu begegnen, ist es notwendig, den Zugriff auf unterschiedliche Visualisierungsmethoden zu ermöglichen und gleichzeitig den Aufwand für die Integration von Abbildungsverfahren in eine interaktive Anwendung weitgehend zu reduzieren. Systeme, welche dem Anwender in Bezug auf dessen Visualisierungsziel den Einsatz von linearen und nichtlinearen Darstellungsweisen bieten, werden bisher im Umfeld interaktiver 3D-Anwendungen kaum eingesetzt. Die Bereitstellung einer softwaretechnischen Basis vermag einerseits die Analyse und andererseits den anwendungsbezogenen Einsatz von Visualisierungstechniken in 3D-Benutzerschnittstellen – isoliert und kombiniert – sicherzustellen.

In diesem Kapitel werden die Anforderungen an eine Forschungs- und Entwicklungsplattform spezifiziert, deren Erfüllung eine Integration von Darstellungsverfahren in den Entwurfsprozess von 3D-Benutzerschnittstellen sicherstellt und den Einsatz von linearen und nichtlinearen Darstellungsformen im interaktiven Prozess von Anwender und System ermöglicht. (4.1). Die Ausführungen in Abschnitt 4.2 werden zeigen, dass etablierte Komponententechnologien aus der Softwaretechnik einen zielführenden Lösungsansatz für die Umsetzung bieten. Etablierte *Software-Komponentenmodelle* werden dazu hinsichtlich deren Eignung für die Entwicklung interaktiver 3D-Anwendungen betrachtet und die in den Komponentenmodellen fehlende Unterstützung der 3D-Computergrafik durch die Integration einer *Grafik-Engine* kompensiert. Die praktische Umsetzung ist die interaktive Arbeitsumgebung »Bildsprache LiveLab (BiLL)«, die an der *Professur für Mediengestaltung der Technischen Universität Dresden* entwickelt wird. BiLL wird in Abschnitt 4.3 vor dem Hintergrund erläutert, einerseits als Plattform zur Integration von Visualisierungsbausteinen, andererseits als ein Werkzeug zur experimentellen Gestaltung interaktiver 3D-Benutzerschnittstellen zu fungieren.



## 4.1 Anforderungen und allgemeine Vorgehensweise

### 4.1.1 Ansprüche und Herausforderungen

Im dritten Kapitel dieser Arbeit wird aufgezeigt, dass die Bestrebungen im Bereich der Mensch-Maschine-Kommunikation zu einer fortwährenden Entwicklung und Verbesserung von 3D-Benutzerschnittstellen führt und in den zurückliegenden zwei Jahrzehnten eine große Anzahl von interaktiven Anwendungen und Visualisierungssystemen hervorbrachte. Die Vielfalt der Softwaresysteme erstreckt sich von high-level Bibliotheken, beispielsweise *OpenSG* [REINERS U. A. 2002], *OGRE* [OGRE] und *OpenSceneGraph* [WANG & QIAN 2010] bis zu vollständigen 3D-Anwendungen, beispielsweise *Alice* [ALICE], *Open Inventor* [WERNECKE 1994] und *MeVisLab* [MEVISLAB]. Der Fokus von 3D-Anwendungen liegt dabei vorwiegend auf spezifischen Teilbereichen einer Anwendungsdomäne. So ist Alice eine Programmierumgebung für die Erstellung von 3D-Animationen, Open Inventor ein Werkzeug für den computergestützten Entwurf und MeVisLab ein Forschungs- und Entwicklungssystem für die medizinische Bildverarbeitung und Visualisierung. Im Bereich des Non-Photorealistic Rendering existieren vor allem prototypische 3D-Anwendungen. Deren Entwicklung erstreckt sich oftmals über einen lediglich begrenzten Zeitraum, da diese Softwaresysteme als Demonstratoren dienen oder über die Umsetzung eines konkreten Untersuchungsgegenstandes hinaus, nicht fortgeführt werden. Da Entwicklungen im Bereich des Non-Photorealistic Rendering weiterhin notwendig sind, ist eine Plattform für die Forschung und Entwicklung im Bereich der experimentellen Interfaceentwicklung bereitzustellen. Für die Realisierung einer 3D-Anwendung mit dem Ziel das große Repertoire an Darstellungsweisen in einer interaktiven 3D-Anwendung zur Verfügung zu stellen, ist zunächst die Frage zu beantworten, in welcher Form die Darstellungsverfahren zu integrieren sind und welche softwaretechnologische Basis hierzu geeignet ist. Dabei steht der Anspruch im Vordergrund, die linearen und nichtlinearen Projektionsverfahren plattformunabhängig und wiederverwendbar bereitzustellen. In diesem Zusammenhang konstatiert BOWMAN, dass die 3D-Anwendungsdomäne sehr von Komponentenbausteinen profitiert kann (vgl. [BOWMAN U. A. 2006, S. 7]). Gleichzeitig wird in [WINGRAVE 2008, S. 13] jedoch aufgezeigt, dass bisher kaum wiederverwendbare Softwarekomponenten für Visualisierungsverfahren und Interaktionstechniken in interaktiver 3D-Anwendungen eingesetzt werden. Dabei verknüpft WINGRAVE seine Aussage mit einer Erwartungshaltung in Bezug auf die Entwicklungen im Bereich interaktiver 3D-Anwendungen: „*The hope is that 3D systems will begin to behave similarly and development will be simpler because of a standard set of components.*“<sup>53</sup> [WINGRAVE 2008, S. 14–15]

Im Folgenden werden Anforderungen in Bezug auf die Konstruktion von 3D-Benutzerschnittstellen aufgestellt, die eine komponentenbasierte Arbeitsumgebung idealerweise erfüllen sollte. Die Forderungen an die 3D-Anwendung, die auszugsweise in [WOJDZIAK U. A. 2011A, S. 260] beschrieben werden, besitzen einen undogmatischen Charakter. Eine Invariabilität würde die notwendige Anpassungsfähigkeit bei der Umsetzung einer interaktiven Anwendung mit einem experimentellen Charakter vorschnell eingrenzen. Die Beschränkung auf die

<sup>53</sup> Die Hoffnung ist, dass 3D-Systeme beginnen sich ähnlich zu verhalten und die Entwicklungen vereinfacht werden, aufgrund eines Standard-Sets von Komponenten

Darstellung dreidimensionaler Inhalte, wie sie bereits in der Abgrenzung im ersten Kapitel der Arbeit beschrieben wird, besteht jedoch. Weiterhin stellt eine *Forschungs- und Arbeitsumgebung* innerhalb der definierten Rahmenbedingungen einen Kompromiss zwischen der zielgerichteten Bedienbarkeit in einem konkreten Anwendungskontext und der Flexibilität im Untersuchungskontext dar.

### 4.1.2 Generelle Anforderungen

- *Einsatz von Szenengraphen:* Die Nutzung des bewährten Szenengraphkonzepts für die Strukturierung und Verarbeitung der Datenbasis interaktiver 3D-Benutzerschnittstellen ist erforderlich. Hierbei sollte der Einsatz eines weitverbreiteten oder standardisierten 3D-Formats beziehungsweise einer Grafik-Bibliothek auf Graphenbasis als Realisierungsgrundlage eingesetzt werden.
- *Anwendungsvielfalt:* Die Verwirklichung von 3D-Benutzerschnittstellen für vielfältige Anwendungsszenarien ist im Interaktionsprozess mit der Arbeitsumgebung wesentlich. Demzufolge sollte nicht nur die Integration der Abbildungsverfahren, sondern ebenso die Unterstützung des interaktiven Prozesses für die Parametrisierbarkeit und Adaptierbarkeit der Verfahren mithilfe einer grafischen Benutzerschnittstelle gewährleistet sein.
- *Erweiterbarkeit zur Laufzeit:* Die Architektur des Softwaresystems muss unterschiedliche Visualisierungsziele und Anwendungsszenarien mit veränderlichen Anforderungen und Funktionsumfängen unterstützen. Daher ist eine hohe Flexibilität notwendig, die durch eine Veränderung des Funktionsumfangs während der Laufzeit erreicht werden kann.
- *Plattformunabhängigkeit:* Der Einsatz der Anwendung auf unterschiedlichen Betriebssystemen ist erforderlich, um die Applikation einem möglichst großen Benutzerkreis zur Verfügung zu stellen. Infolgedessen ist die Unterstützung der gängigsten Plattformen zu gewährleisten.
- *Echtzeitfähigkeit:* Bei einer Arbeitsumgebung im Kontext interaktiver Visualisierungen muss die Echtzeitfähigkeit für die Darstellung von Animationen sowie für eine verzögerungsfreie Interaktion gewährleistet sein. Die Darstellung in Echtzeit sollte auch gegeben sein, wenn rechenintensive Abbildungsverfahren, beispielsweise unter Einsatz mehrerer computergrafischer Kameras, zu einem hohen Berechnungsaufwand führen.

### 4.1.3 Anforderungen an Visualisierungsbausteine

Im Folgenden werden Anforderungen zur Verwirklichung eines modularen Ansatzes für die einfache Konstruktion von 3D-Benutzerschnittstellen aufgestellt, welche die Softwarearchitektur und die Softwarekomponenten der Arbeitsumgebung idealerweise erfüllen sollte. In der Anforderungsanalyse werden wiederverwendbare Visualisierungsbausteine allgemein als Komponenten bezeichnet.

- *Die Konfigurierbarkeit von Komponenten:* Für die Bereitstellung einer vielseitigen Funktionalität ist die Parametrisierung jeder Visualisierungskomponente erforderlich.

Durch eine werkzeuggestützte Modifizierbarkeit der Parameter wird eine Nutzung der Arbeitsumgebung in verschiedenen Benutzerkreisen sichergestellt. Zur Wahrung der experimentellen Ausrichtung der Arbeitsumgebung sollten die Manipulationsmöglichkeiten jedoch per se nicht aufgabenspezifisch sein.

- *Die Komposition und Gruppierung:* Die Visualisierungsbausteine müssen über Schnittstellen verfügen, um miteinander verknüpft werden zu können. Dazu sollten die Komponenten einerseits einen wohl definierten Funktionsumfang bieten, andererseits formal definierte Dienste zur Kommunikation mit anderen Komponenten der Arbeitsumgebung nutzen.
- *Deskriptive Komponentenschnittstellen:* Für den effektiven Einsatz der Arbeitsumgebung sowie jeder Komponente ist eine strukturierte, textuelle Beschreibung der gebotenen Funktionalität, der konfigurierbaren Parameter, der Abhängigkeiten und der Komponentenausrichtung notwendig. Darüber hinaus ist die Verfügbarkeit von Metainformationen zu den Eigenschaften und der Verwendung, der Version und dem Veröffentlichungsdatum, zweckmäßig.
- *Die Trennung von Erstellung und Verwendung:* Die Unterstützung in den verschiedenen Stufen der Entwicklung und Anwendung von Visualisierungsbausteinen sollte durch die Bereitstellung von Werkzeugen, 3D-Daten und Dokumentationen erfolgen. Hierbei ist zwischen der Komponenten- und der Anwendungsentwicklung sowie der Verwendung zur Laufzeit des Systems zu differenzieren.

#### 4.1.4 Anforderungen an eine Arbeitsumgebung

Um den interaktiven Prozess des Nutzers – speziell zur Erzeugung und Parametrisierung von Visualisierungsmethoden sowie deren Einbindung in den Schnittstellenentwurf – zu unterstützen, sollten die nachfolgend formulierten Anforderungen durch die Anwendung erfüllt werden:

- *Gestaltung des Bildraums:* Die Arbeitsumgebung sollte grafische Modifikationsmöglichkeiten bereitstellen, die eine parametrisierte Veränderung von algorithmischen Verfahrensweisen ermöglicht. Dieser interaktive Prozess zur Erstellung eines Bildraums sollte auf Basis der aufgezeigten Manipulationsmöglichkeiten des computergrafischen Kameramodells getätigt werden können.
- *Automatische Bilderzeugung:* Die automatische Ausführung von Visualisierungsverfahren sollte möglich sein, wobei durch eine vordefinierte Konfiguration eines Verfahrens eine möglichst gebrauchstaugliche Darstellung erreichen wird. Ferner sollte jedoch eine manuelle Rekonfiguration über die grafische Benutzerschnittstelle durchführbar sein.
- *Kontextuelle Beschreibung:* Die Bereitstellung einer Abstraktionsebene für die Interfaceentwicklung, die den Entwurf von 3D-Interfaces auf Basis von Nutzeraufgaben und Visualisierungszielen ermöglicht, ist anstelle einer Spezifikation und Konfiguration von computergrafischen Renderdetails notwendig.

- *Adaptierbare Benutzerschnittstelle*: Unterschiedliche Personenkreise sind in die Erforschung und Entwicklung von Benutzerschnittstellen interaktiver 3D-Systeme involviert. Eine Subsumierung von Anwendern in Benutzergruppen (beispielsweise Computergrafiker, Interface- und Interaktionsdesigner) ist daher zweckmäßig. Eine benutzerzentrierte Gestaltung der Arbeitsumgebung umfasst eine angepasste Bedienbarkeit und erfüllt die Grundsätze der Dialoggestaltung.

## 4.2 Basistechnologien des Bildsprache LiveLab

Eine Softwarearchitektur für die Entwicklung und interaktive Nutzung von 3D-Benutzerschnittstellen stellt hohe Ansprüche an die Grafikverarbeitung des Computers. Die computergrafische Bildsynthese muss mithilfe der Grafikhardware effizient berechnet werden, um die Verarbeitung und Darstellung des Objektraums auf dem Ausgabemedium in Echtzeit zu ermöglichen. Damit die Erzeugung eines 3D-Interface und die Anpassung des Abbildungsprozesses nicht auf Basis hardwarenaher Grafikbefehle zu spezifizieren sind, werden *Grafik-APIs* eingesetzt, die eine direkte Schnittstelle zur Grafikhardware bereitstellen. Auf diesen Schnittstellen, insbesondere OpenGL und Direct3D, bauen *3D-Grafik-Engines* auf und stellen abstrahierte und erweiterte Bildsynthese-Funktionen auf einer höheren Abstraktionsebene zur Verfügung. Die Funktionalität der 3D-Grafik-Engines ist auf die Verarbeitung und Darstellung von räumlichen Visualisierungen ausgerichtet. Eine interaktive Einflussnahme auf den Abbildungsprozess mithilfe einer grafischen Benutzeroberfläche zur Festlegung von Bildsyntheseparameter ist dabei nur eingeschränkt gegeben. Die 3D-Grafik-Engines bieten zwar eine Auswahl elementarer Funktionen für die Interaktion zwischen dem Anwender und dem computergrafischen Bildraum, der Umfang und die Funktionalität von grafischen Werkzeugen zur Spezifikation und Bearbeitung des Abbildungsprozesses sind jedoch begrenzt.

Die Erfüllung der Anforderungen an die Arbeitsumgebung macht es erforderlich, die Verwaltung von Programmfenstern und die Verfügbarkeit von Kontroll- und Steuerungselementen zu realisieren. Die dafür benötigten Funktionen sind aufgrund der Nähe zum zugrunde liegenden Betriebssystem häufig sehr systemspezifisch und komplex. In der Folge ist ein umfangreicher Implementierungsaufwand gegeben. Eine Alternative sind plattformunabhängige *GUI-Toolkits*. Diese Toolkits sind Sammlungen von Bibliotheken und Schnittstellen, die das Erstellen und das Gestalten von konventionellen 2D-Benutzeroberflächen durch die Bereitstellung von grafischen Bedienelementen und Standardfunktionalitäten vereinfachen und die Entwicklung von Anwendungsoberflächen vereinfachen. Hierdurch können plattformübergreifende 2D-Benutzerschnittstellen verwirklicht werden, welche die Interaktionsmöglichkeiten innerhalb einer 3D-Anwendung erweitern.

Darüber hinaus ist es notwendig, ein softwaretechnisches Fundament für die Entwicklung und Ausführung von Visualisierungsbausteinen, unter Berücksichtigung der Anforderungen nach einer dynamischen Erweiterbarkeit und einer flexiblen Anpassbarkeit, bereitzustellen. Dies beinhaltet zugleich strukturelle Vorgaben in Bezug auf die Kommunikationsmechanismen und die verteilte Programmierung sowie die Regelung von Verknüpfungs- und Kompositionsmöglichkeiten hinsichtlich der Dienste und Schnittstellen. Komponentenarchitekturen

sind ein probates Instrument, die Ausgestaltung einer interaktiven Anwendung und die Integration von Methoden und Verfahren auf Basis einer Kompositionssprache zu definieren und zu regulieren. Die Einbindung einer Komponentenarchitektur bietet in der Folge eine dynamische Softwarebasis, wodurch die Modularisierung der Arbeitsumgebung erreicht wird.

Ein weiterer zu berücksichtigender Aspekt bei der Entwicklung einer komponentenbasierten 3D-Anwendung ist die Festlegung der Programmiersprache. Im Bereich der echtzeitfähigen 3D-Grafik bietet die Programmiersprache C++ hervorragende Voraussetzungen, da die Grafik-Hardware flexibel ansteuerbar ist und hardwarenahe 3D-Schnittstellen gut integriert werden können (vgl. [STROUSTRUP 2007, S. 29 ff.]). C++ realisiert darüber hinaus das objekt-orientierte Programmierparadigma, das einen hohen Grad an Flexibilität und Wiederverwendung aufweist und die Softwareentwicklung durch Gruppen vereinfacht (vgl. [STROUSTRUP 2000, S. 44 ff.]). Die Festlegung von C++ als Programmiersprache beeinflusst den Einsatz von 3D-Grafik-Engines und GUI-Toolkits. Darüber hinaus hat die Regelung der Programmiersprache Auswirkungen auf die potenziellen Komponentenarchitekturen, da eine einheitliche Programmiersprache eine Voraussetzung für eine effektive Entwicklung der Softwarearchitektur ist. Aufgrund dessen werden die nachfolgenden Technologiebetrachtungen unter Berücksichtigung der Programmiersprache und immer vor dem Hintergrund einer möglichst flexiblen Arbeitsumgebung zur Gestaltung interaktiver 3D-Benutzerschnittstellen erfolgen.

## 4.2.1 Komponentensystem

Im Hinblick auf die in Abschnitt 4.1 aufgestellten Anforderungen bietet die *komponentenbasierte Softwareentwicklung* eine mögliche Lösung für die Verwirklichung einer interaktiven 3D-Arbeitsumgebung ist. Besonders die unter 4.1.3 formulierten Anforderungen an wiederverwendbare Visualisierungsbausteine finden in den Eigenschaften von Softwarekomponenten, wie diese von SZYPERSKI formuliert sind, eine Entsprechung: *“A software component is a unit of composition with contractually specified interfaces and explicit context dependencies only. A software component can be deployed independently and is subject to composition by third parties”*<sup>54</sup> [SZYPERSKI U. A. 1999, S. 34]. Ein Hauptmerkmal von *Softwarekomponenten* ist nach SZYPERSKI deren Kompositionsfähigkeit. Danach ist eine Komponente allgemein betrachtet ein in sich geschlossener, kombinierbarer, wiederverwendbarer Softwarebaustein, der über eine definierte syntaktische Schnittstelle und eine Semantik verfügt. Für eine weiterführende Betrachtung komponentenorientierter Softwareentwicklung sei an dieser Stelle auf die Literatur, unter anderem [SZYPERSKI U. A. 1999] und [ABMANN 2003], verwiesen. In Bezug auf eine interaktive 3D-Arbeitsumgebung werden allerdings Komponentensysteme und deren wesentliche Merkmale in Kurzform dargestellt.

Ein Komponentensystem ist durch drei Charakteristiken beschreibbar: ein *Komponentenmodell*, eine *Kompositionstechnik* und eine *Kompositionssprache* (vgl. [ABMANN 2003, S. 17 ff.]). Das Komponentenmodell beschreibt die Art der Komponente, indem die Funktion, die Struktur und die verfügbaren Schnittstellen des Softwarebausteins definiert werden. Die

<sup>54</sup> „Eine Software-Komponente ist eine Kompositionseinheit mit vertraglich festgelegten Schnittstellen und expliziten Kontextabhängigkeiten. Eine Software-Komponente kann unabhängig eingesetzt werden und unterliegt der Komposition durch Dritte.“

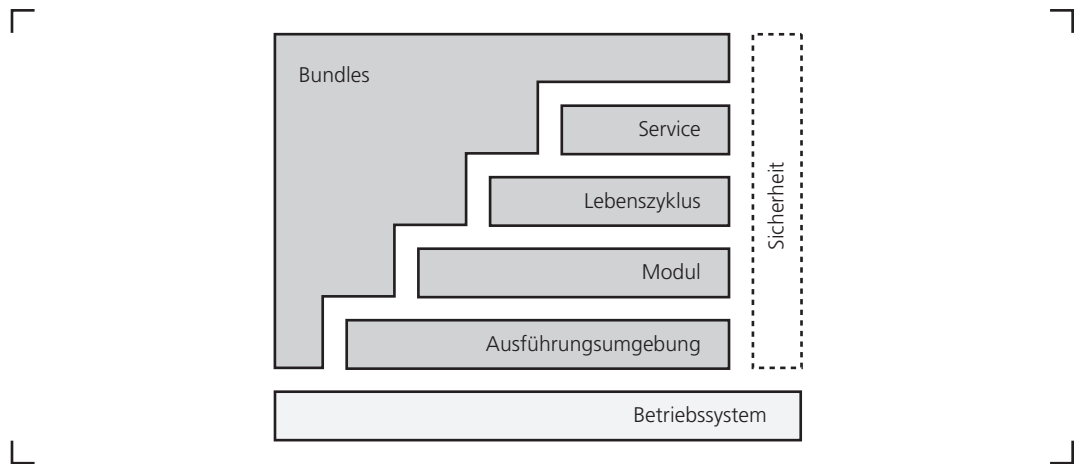


Abbildung 21: Bestandteile der OSGi-Architektur als Schichtenmodell, nach [OSGi ALLIANCE 2007]

Kompositionstechnik legt fest, in welcher Form die Softwarekomponenten miteinander verbunden werden, während die Kompositionssprache zur Steuerung und Parametrisierung dieses Prozesses dient. Komponentenmodelle existieren in verschiedenen Ausprägungen, die eine unterschiedliche Flexibilität und Benutzerfreundlichkeit aufweisen. Sehr flexible Systeme, beispielsweise *Toolbus*, nutzen eine Prozess-Algebra, um Komponenten und ihre Komposition deskriptiv zu definieren (vgl. [DE JONG & KLINT 2003, S. 222 ff.]). Das Komponentenmodell arbeitet mit sogenannten *Tools*, deren Funktionsweise sowie Interaktion mithilfe von abstrakten Datentypen und einer Prozess-Algebra beschrieben wird. Aufgrund der deskriptiven Techniken übernimmt die Komponentenlaufzeit die Verarbeitung der Komponenten und die Organisation der Zusammenarbeit vollständig. Die Kompositionstechnik und die Kompositionssprache sind dabei identisch. Der Toolbus selbst ist eine Koordinationsarchitektur, auf deren Grundlage die Komponenten auf verschiedenen Plattformen und mit Implementierungen in unterschiedlichen Programmiersprachen zusammenarbeiten können. Aufgrund der Ausrichtung der Komponentenarchitektur – eine hohe Flexibilität und Wiederverwendbarkeit – ist die Anwendung der Prozess-Algebra von Toolbus mit großem Aufwand verbunden und bedarf weitgehender Vorkenntnisse eines Programmierers.

Ein weiteres Komponentensystem ist *Common Object Request Broker Architecture*<sup>55</sup> (*CORBA*), das für verteilte Objekte, beispielsweise im Netzwerk oder im Internet entwickelt wird. CORBA ist eine Spezifikation für eine objektorientierte Vermittlungssoftware, deren Kern ein sogenannter Object Request Broker bildet. Dieser definiert die plattformübergreifenden Protokolle und Dienste. CORBA-konforme Implementierungen vereinfachen, auf Basis des Komponentenmodells sowie der spezifizierten Kompositionssprache, das Erstellen verteilter Anwendungen in heterogenen Umgebungen (vgl. [OMG]). Hierbei liegt ein abstraktes Konzept zugrunde, das von der Object Management Group (OMG) entwickelt wird. Dieses ermöglicht es, aufgrund eines hohen Abstraktionsgrades, verteilte Anwendungen mit einem vergleichsweise geringen Arbeitsaufwand zu entwickeln.

<sup>55</sup> *Common Object Request Broker Architecture* engl. Architektur für Vermittler von Objektnachrichten

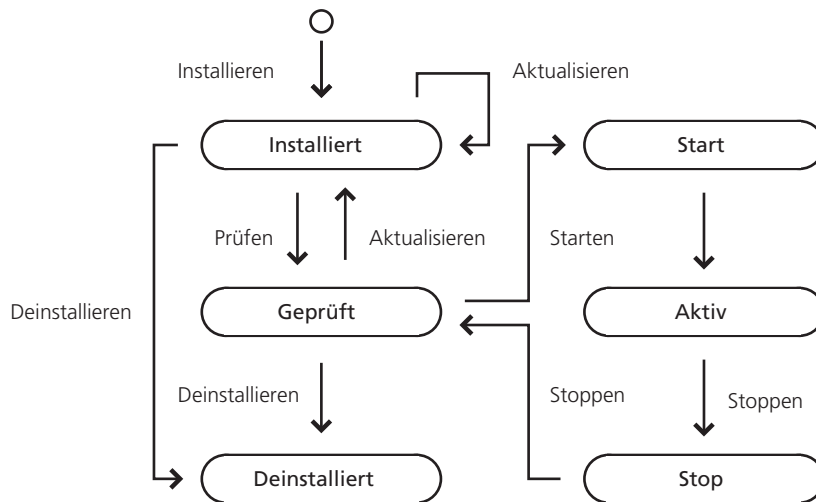


Abbildung 22: Mögliche Zustände von OSGi-Bundles innerhalb des OSGi-Lebenszyklus als Zustandsdiagramm (nach [OSGi Alliance 2007])

Einer der entwicklerfreundlichsten Komponentenstandards wird in der Spezifikation der *Open Services Gateway initiative* (OSGi) beschrieben (vgl. [OSGi ALLIANCE 2007]). OSGi bietet ein Konzept zur Modularisierung von Java-Applikationen und implementiert eine Erweiterung des bestehenden Modulsystems der Programmiersprache Java. Komponenten werden als *Bundles* bezeichnet und werden vorwiegend in komprimierten Archiv-Dateien veröffentlicht und installiert. Diese enthalten Binärdateien und alle notwendigen Ressourcen sowie eine Beschreibung des Bundles. Diese Erläuterung enthält Informationen über die Komponente, insbesondere Abhängigkeiten von anderen Bundles oder Ressourcen. Die OSGi-Architektur besteht aus verschiedenen Schichten, die in der Spezifikation beschrieben und definiert sind. Aufeinander aufbauend besteht das Framework aus den folgenden Schichten: *Ausführungsumgebung*, *Modul*, *Lebenszyklus*, *Service* und *Sicherheit* (siehe Abbildung 21). Die Ausführungsumgebung verwaltet das Laden und Ausführen von Bundles entsprechend der gegebenen Abhängigkeiten. Die Modulschicht spezifiziert das Komponentenmodell durch die Festlegung des Formates der Bundles und deren Distribution sowie den Austausch von Ressourcen. Während in der Modulschicht primär statische Aspekte festgelegt sind, werden in der Lebenszyklusschicht die dynamischen Faktoren spezifiziert. In Abbildung 22 werden die potenziellen Zustände eines Bundles dargestellt. Der Lebenszyklus wird durch die Zustände *Installieren*, *Starten*, *Stoppen*, *Aktualisieren* und *Deinstallieren* definiert. Der Ablauf wird in Abbildung 22 entsprechend der OSGi-Spezifikation dargestellt. Die Serviceschicht dient dazu, vorhandene Bundles im OSGi-Framework zu registrieren und den Austausch von Funktionalitäten zwischen den Bundles sicherzustellen. Dies ist essenziell für die Interoperabilität der Komponenten, da durch die Services eine definierte und spezifizierte Beschreibung der Bundles und deren Funktionalität bereitgestellt werden. Weiterhin bildet dies die Grundlage für die Komposition der Komponenten. Die Sicherheitsschicht erweitert das Sicherheitsmodell von Java. Mit den Mechanismen dieser Schicht lassen sich die Ausführungsrechte einzelner Bundles gezielt einschränken. Die Sicherheitsschicht bietet hierfür die Möglichkeit, Module nach dessen Herkunft oder mittels einer digitalen Signatur zu authentifizieren. Diese Schicht



ist innerhalb der OSGi-Architektur optional. Die Bundles selbst befinden sich in der Bundle-Schicht, welche auf den bereits genannten Schichten aufbaut.

Eine vollständige Implementierung des OSGi-Standards in der vorgesehenen Programmiersprache C++ ist nicht möglich, weil die Spezifikation explizit auf Eigenschaften der Programmiersprache Java beruht. Insbesondere Elemente der Sicherheitsschicht können in anderen Programmiersprachen nicht umgesetzt werden. Der portierbare Teil des ursprünglichen OSGi-Standards wird jedoch durch das *Service Oriented Framework* (vgl. [SOF]) sowie die *Open Service Platform (OSP)* (vgl. [APPLIED INFORMATICS]) implementiert. Letztere ist eine kommerzielle Umsetzung und derzeit das umfassendste OSGi-Framework in C++. OSP basiert auf den frei erhältlichen *POCO C++ Bibliotheken* [APPLIED INFORMATICS]. Diese stellen eine plattformunabhängige Standardbibliothek für C++ bereit, die dem Funktionsumfang der Standardbibliotheken von Java und .NET entspricht. Die Open Service Platform implementiert einen vergleichsweise vollständigen Lebenszyklus von Bundles, verzichtet jedoch weitestgehend auf die Sicherheitsschicht.

### 4.2.2 Grafik-Engine

Zu Beginn der Entwicklungen computergrafischer Anwendungen war deren Implementierung an Spezialhardware gebunden. Erst durch die Einführung von Grafik-APIs wurden Software-schnittstellen zur Grafikhardware bereitgestellt, die eine weitgehende Unabhängigkeit von den zugrunde liegenden Computerkomponenten boten. Derzeit finden vornehmlich zwei Spezifikationen von Grafikschnittstellen in interaktiven 3D-Anwendungen eingesetzt; einerseits die *Open Graphics Library (OpenGL)*, andererseits *Direct3D*. Die Letztgenannte ist eine plattformabhängige Programmierschnittstelle der Firma Microsoft, die vorwiegend Anwendung in Computerspielen findet. Die API konkurriert mit der plattform- und betriebssystemunabhängigen Grafikbibliothek OpenGL. Diese Grafik-API ist ein offener Standard, die eine große Verbreitung im Bereich professioneller Grafikanwendungen für Wissenschaft, Medizin und Wirtschaft aufweist, aber auch in der Unterhaltungsindustrie eingesetzt wird. Beide Schnittstellen verfügen über einen annähernd identischen Funktionsumfang und sind hinsichtlich der Performanz in computergrafischen Anwendungen vergleichbar. Durch die Spieleindustrie, die überwiegend Anwendungsentwicklungen für das Microsoft Betriebssystem vollzieht und nach fotorealistischer Computergrafik strebt, werden die neuesten Entwicklungen zunächst in Direct3D umgesetzt. Eine Integration erfolgt ebenso, jedoch zeitlich verzögert, in OpenGL. Um die gegebene Anforderung der Plattformunabhängigkeit zu erfüllen, ist OpenGL der Grafik-API von Microsoft vorzuziehen. Aufgrund dessen sind die weitergehenden Betrachtungen auf OpenGL basierende und in der Programmiersprache C++ umgesetzte 3D-Grafik-Engines ausgerichtet.

Die Anforderungen im Unterabschnitt 4.1 legen weiterhin fest, dass für eine Umsetzung von Darstellungstechniken in der 3D-Arbeitsumgebung einerseits die Visualisierung des Objektraums verwirklicht sein muss, andererseits die visuelle räumliche Struktur auf Szenengraphbasis verwaltet werden soll. Moderne *Szenengraph-APIs* kombinieren die Implementierung des Szenengraphmanagements mit den Aufgaben der Bildsynthese, wodurch diese Grafik-Toolkits die Lücke zwischen komplexer Szenenbeschreibung und der Bildsynthese auf

eine komfortable Weise schließen. Als Teil eines Anwendungssystems, das für die computergrafische Darstellung entscheidend ist, werden die Toolkits häufig synonym als Grafik-Engines bezeichnet (vgl. [SOWIZRAL 2000, S. 57]). Die Szenengraph-APIs können als Grafik-Engines mit hierarchischem Objektmanagement betrachtet werden, grenzen sich dabei von den zusätzlich verfügbaren *Spiele-Engines* ab. *Spiele-Engines* sind umfangreiche und berechnungseffiziente 3D-Toolkits inklusive eines Szenengraphmanagements, die vor allem durch die Entwicklung von Computerspielen mit aufwendiger 3D-Grafik im Unterhaltungsbereich eine große Verbreitung erreichen. Die *Spiele-Engines* können ebenfalls als Grundlage für computergrafische Ausgaben verwendet werden, wobei diese gegenüber Szenengraph-APIs über zusätzliche Elemente, beispielsweise Audiosystem, künstliche Intelligenz, Spiellogik und die Simulation physikalischer Vorgänge, verfügen. Szenengraph-APIs hingegen integrieren diese Leistungsmerkmale für gewöhnlich nicht, sind aber flexibel einsetzbar und durch eine hohe Interoperabilität mit Programmbausteinen interaktiver Anwendung charakterisiert. Die Szenengraph-APIs können durch die gegebene Flexibilität kontextspezifisch angepasst und in der Folge für verschiedenartige Einsatzgebiete genutzt werden. Durch die Ausrichtung einer 3D-Anwendung als Arbeitsumgebung, mit dem Fokus auf bildstrukturelle Aspekte einer Benutzerschnittstelle, finden die zusätzlichen Elemente der *Spiele-Engines* keine Anwendung. In der Folge können Szenengraph-APIs die gestellten Anforderungen zielgerichteter erfüllen.

Es gibt keine einheitlichen Standards für die Funktionsweise und die Schnittstellen von 3D-Grafik-Engines. Vielmehr existiert eine Reihe von etablierten Toolkits und Frameworks, die für spezifische Anwendungsgebiete konzipiert sind. Beispielsweise ist *Unity* eine 3D-Grafik-Engine deren Einsatz für mobile Geräte, Spielkonsolen und Web-Browser optimiert ist. Die Grafik-Engine *OGRE3D* wird für den Anwendungsbereich der konventionellen Computerspiele konzipiert sowie entwickelt und bietet eine Schnittstelle zu Direct3D und OpenGL beinhaltet. Der bereitgestellte Funktionsumfang und vor allem die Unterstützung von beiden Grafikschnittstellen werden hierbei durch eine Kapselung von Funktionseinheiten erreicht. Dies vereinfacht die Anwendungsentwicklung, schränkt jedoch ein direktes Eingreifen des Entwicklers in die Bildsynthese ein und beeinträchtigt in der Folge die Anpassbarkeit eines 3D-Systems.

*OpenSceneGraph (OSG)* ist eine Grafik-Engine, die überwiegend im Bereich der wissenschaftlichen Visualisierungen und Simulationen eingesetzt wird. OSG ist ein objektorientiertes Grafik-Framework, das auf OpenGL aufbaut und den Entwickler von der Umsetzung und Optimierung von hardwarenahen Grafik-Aufrufen, durch einfache und nachvollziehbare Schnittstellen, befreit (vgl. [WANG & QIAN 2010, S. 10 ff.]). Im Gegensatz zu *OGRE3D* ist ein Eingreifen in die Bildsynthese über bereitgestellte Schnittstellen komfortabel möglich, wodurch eine Anpassung der Abbilderzeugung vereinfacht wird. Darüber hinaus enthält OSG weitere Dienstprogramme für die schnelle Entwicklung von Grafik-Anwendungen im Kontext von wissenschaftlichen Visualisierungen und Simulationen, weshalb sich diese Grafik-Engine für den Einsatz in einer Arbeits- und Experimentierumgebung eignet.

### 4.2.3 Benutzeroberflächen-Toolkit

Zur Unterstützung der Mensch-Computer-Kommunikation in einer interaktiven Arbeitsumgebung, im Besonderen für die experimentelle Modifikation der computergrafischen Bildsynthese, ist zusätzlich zum visualisierten Bildraum eine konventionelle 2D-Benutzeroberfläche ein effektives Werkzeug. Für eine faktische Parametrisierung der Bildsynthese sind schnell ausführbare, inkrementelle und reversible Benutzeraktionen notwendig, welche ein unmittelbares Feedback im Bildraum erzeugen. Da die Benutzervorgaben weder durch das Komponentenmodell noch durch die Grafik-Engine effizient verwirklicht werden können, ist eine weitere Bibliothek, ein *GUI-Toolkit*, notwendig. Plattformübergreifende Toolkits für grafische 2D-Benutzeroberflächen fassen die verschiedenen betriebssystemspezifischen Befehle in Funktionen zusammen und verbergen die Eigenheiten der zugrunde liegenden Systeme vor dem Entwickler. Dadurch kann eine Benutzeroberfläche unabhängig vom verwendeten Betriebssystem entworfen und umgesetzt werden. Neben der Unterstützung mehrerer Zielplattformen ist die Implementierung in C++ eine Voraussetzung für den Einsatz in der Arbeitsumgebung. Die Bibliothek sollte eine Zusammenstellung verschiedener grafischer Benutzeroberflächenelemente bieten, damit die Einbindung einer Interaktionsvielfalt in das Interface erfolgen kann.

Der Erfinder von C++, BJARNE STROUSTRUP, zählt in seinem Bericht 25 verfügbare Bibliotheken für grafische Benutzerschnittstellen auf (vgl. [STROUSTRUP 2007, S. 28]). *Qt*, *wxWidgets* und *FLTK* sind drei Toolkits in dieser Aufzählung, die in verschiedenen Anwendungsdomänen weit verbreitet sind. *Qt* ist ein sehr umfangreiches C++-Toolkit (vgl. [BLANCHETTE & SUMMERFIELD 2008, S. 18 ff.]), das zusätzlich zu Interfaceelementen umfangreiche Funktionen zur Internationalisierung sowie Datenbankfunktionen und XML-Unterstützung bietet. Trotz umfassender Dokumentationen gestaltet sich aufgrund des Funktionsumfangs der Einstieg in die Bibliothek für den Entwickler aufwendig. Die für den gegebenen Kontext zu große Funktionsvielfalt birgt für die Implementierung einer 3D-Anwendung den Nachteil, dass unerfahrene Entwickler den Überblick über die Struktur verlieren und die zielgerichtete Nutzung der verfügbaren Klassen und Methoden nicht verwirklichen können. Aufgrund dessen eignet sich *Qt* nicht für die gegenständliche Arbeitsumgebung. *Fast Light Toolkit* (FLTK) ist eine performante, einfach zu verwendende Bibliothek (vgl. [FLTK]). Das GUI-Toolkit ist kompakt und auf die Bereitstellung der wesentlichen Funktionalitäten beschränkt. Dadurch ist FLTK einfach und flexibel zu verwenden. Unter generellen Gesichtspunkten zeichnet sich das Toolkit, aufgrund sorgfältiger Optimierung, durch einen sehr geringen Ressourcenbedarf und eine hohe Geschwindigkeit aus. In der Folge ist FLTK für den Einsatz in einer Arbeitsumgebung, mit dem Schwerpunkt auf eine rechenintensive Darstellung von 3D-Szenen, geeignet. Ein weiteres GUI-Toolkit ist *wxWidgets*. Die plattformunabhängige Klassenbibliothek verwendet betriebssystemeigene Kontrollelemente und Dienstprogramme, um ein betriebssystemabhängiges *Look-and-Feel*<sup>56</sup> zu erzeugen (vgl. [SMART u. A. 2006, S. 540]). Das bedeutet, dass *wxWidgets* für die Darstellung weitestgehend auf die Standardelemente eines Betriebssystems zugreift und diese bei Bedarf mit eigenen Kontrollelementen ergänzt. So werden plattformnative Funktionalitäten abstrahiert und lediglich fehlende Bedienelemente neu implementiert. Zu den prominentesten Anwendungen, welche auf der Basis von *wxWidgets*

<sup>56</sup> Der Fachbegriff bezeichnet standardisierte Design-Aspekte grafischer Benutzeroberflächen

entwickelt werden, zählen der *VLC Media Player* [VLC] und der Audioeditor *Audacity* [AUDACITY-TEAM]. Das GUI-Toolkit besitzt einen beachtlichen Funktionsumfang und umfasst eine Vielzahl von Klassen, die eine Erstellung einer flexiblen und adaptierbaren Benutzeroberfläche begünstigt. In der Folge erfüllt wxWidgets die gegebenen Anforderungen und bietet ferner die besten Voraussetzungen für eine Einbindung in die Arbeitsumgebung.

#### 4.2.4 Technologieauswahl für die Arbeitsumgebung

An dieser Stelle werden die Basistechnologien in Bezug auf die Anforderungen reflektiert. Im Zuge dessen wird darüber hinaus aufgezeigt, inwieweit das Ineinandergreifen der Bibliotheken und Toolkits zur Realisierung einer Plattform für die Entwicklung und den Einsatz linearer und nichtlinearer Darstellungstechniken im 3D-Interfaceentwurf eine Notwendigkeit darstellt.

Die Herausforderungen im Entwicklungsprozess interaktiver 3D-Anwendungen sind vielfältig und ergeben sich aus den Kontextfaktoren und Anforderungen eines interaktiven Nutzer-Bild-Dialogs. Zu diesen zählen unter anderem die Echtzeitfähigkeit, die kontinuierliche Interaktionsbereitschaft und die Koordination der hohen Anzahl an Freiheitsgraden bei der Interaktion des Nutzers mit einer interaktiven 3D-Anwendung (vgl. [JACOB U. A. 1999, S. 3–4]). Um den Herausforderungen im Entwurf und in der Entwicklung von 3D-Benutzerschnittstellen zu begegnen, stellt WINGRAVE heraus, dass der Softwareentwicklungsprozess auf Basis wiederverwendbarer Softwarekomponenten für die Weiterentwicklung und Verbreitung von 3D-Benutzerschnittstellen wesentlich ist (vgl. [WINGRAVE 2008, S. 15]). Ferner motiviert WINGRAVE die Bestrebungen in diesem Bereich wie folgt: *“A new approach to 3D design and development should focus its efforts not on content, which is being standardized, or behaviors and interaction, which is being explored, but on the representation and architecture of 3DUIs”*<sup>57</sup> [WINGRAVE 2008, S. 34]. Gestützt wird die Aussage WINGRAVES ferner durch die Ergebnisse einer Vielzahl von aktiven Forschungs- und Arbeitsgruppen (vgl. [SHAER U. A. 2008; DACHSELT U. A. 2007]) im Bereich der 3D-Benutzerschnittstellenentwicklung.

In Bezug auf den Entwurf linearer und nichtlinearer 3D-Benutzerschnittstellen in der interaktiven Arbeitsumgebung Bildsprache LiveLab sei an dieser Stelle konstatiert:

- Die Forderungen nach *Erweiterbarkeit zur Laufzeit, Komposition und Gruppierung, Trennung von Erstellung und Verwendung* sowie *deskriptive Benutzerschnittstellen* legen eine hardwareunabhängige dynamische Softwareplattform nahe, wie diese durch die OSGi Plattform, respektive der POJO Open Service Plattform bereitgestellt wird.
- Der Fokus auf den *Einsatz von Szenengraphen, die automatische Bilderzeugung, die Echtzeitfähigkeit* und die *Gestaltung des Bildraums* bewirken, dass keine funktionsreiche, jedoch spezielle Spiele-Engine, sondern eine flexibel einsetzbare Grafik-Engine verwendet wird. Dies bietet die Möglichkeit zur Integration in die Komponentenarchitektur und unterstützt eine flexible Entwicklung von 3D-Benutzerschnittstellen. Durch die Grafik-Engine OpenSceneGraph werden die Forderungen erfüllt.

<sup>57</sup> „Ein neuer Ansatz für den Entwurf und die Entwicklung von 3D-Benutzerschnittstellen sollte die Bemühungen nicht auf den Inhalt, der bereits standardisiert ist, oder auf das Verhalten und die Interaktion, die bereits erforscht ist, sondern auf das Erscheinungsbild und den Aufbau von 3D-Benutzerschnittstellen fokussieren.“

- Die Forderungen nach *Konfigurierbarkeit von Komponenten*, *kontextueller Beschreibung* und *Adaptierbarer Benutzerschnittstellen* werden durch den Funktionsumfang einer 3D-Grafik-Engine nicht erfüllt. Dies bedingt die Einbindung des GUI-Toolkits wxWidgets, da dieses die notwendigen grafischen Interfaceelemente für die Benutzerschnittstelle der Arbeitsumgebung bietet.

Mit dem Ziel der Entwicklung einer Softwareplattform und Arbeitsumgebung ist das Bildsprache LiveLab eine auf existierenden Prinzipien, Methoden und Werkzeugen basierende Lösung für die Analyse und den Einsatz von Darstellungstechniken zur Gestaltung von 3D-Benutzerschnittstellen. Die Plattform nutzt dazu moderne Möglichkeiten der komponentenbasierten Softwareentwicklung durch eine Implementierung der OSGi-Spezifikation unter Verwendung der Programmiersprache C++. Dies ermöglicht ein serviceorientiertes Funktionsprinzip einer komponentenbasierten 3D-Anwendung. Mit der OSGi-Spezifikation können verschiedene Visualisierungstechniken parallel oder aufeinander aufbauend implementiert und zusammenwirkend verwendet werden. Die Interaktion des Nutzers mit dem System wird durch die Einbindung eines GUI-Toolkits zusätzlich unterstützt. Das Bildsprache LiveLab unterliegt einem fortwährenden Entwicklungs- und Erweiterungsprozess durch die Mitarbeiter der Professur Mediengestaltung der Technischen Universität Dresden, basierend auf der Architektur der 3D-Anwendung, die nachfolgend vorgestellt wird.

## 4.3 Die Architektur des Bildsprache LiveLab

In der Umsetzung einer interaktiven 3D-Anwendung, die eine flexible Anpassung des Abbildungsprozesses bietet, wird im Bildsprache LiveLab das etablierte Prinzip der Rendering-Pipeline (siehe Unterabschnitt 3.3.1) mit den Konzepten einer komponentenbasierten Softwareentwicklung kombiniert. Die Rendering-Pipeline ist dabei der Manipulationsgegenstand, deren Modifikation auf Basis von Softwarekomponenten erfolgt. Neben dem konkreten Einsatz und dem Zusammenwirken der Softwarebibliotheken wird die Architektur des Bildsprache LiveLab mit deren wesentlichen Schnittstellen sowie dem Datenaustausch der einzelnen Komponenten auf einem allgemeinen Abstraktionsniveau erläutert. Hierbei werden die Visualisierungskomponenten und deren Zusammenwirken mithilfe von Diensten betrachtet, die einen Einfluss auf die Bildsynthese ausüben, während weiterführende Informationen zum softwaretechnischen Aufbau des Bildsprache LiveLab in [KAMMER U. A. 2012] und auf der Projektwebseite [AG-TV] dargelegt werden.

### 4.3.1 Abbildungsverfahren als Softwarekomponenten

Im Bildsprache LiveLab werden lineare und nichtlineare Abbildungsverfahren als Softwarekomponenten bereitgestellt. Dabei wird in der Umsetzung und der Ausgestaltung eine Trennung zwischen der softwaretechnischen Implementierung der Darstellungsverfahren und dem parametrisierten Eingriff in den Visualisierungsprozess über die grafische Schnittstelle vollzogen. Dabei werden die Anforderungen an die Softwarebausteine sowohl aus Anwender- als auch aus Entwicklersicht berücksichtigt. Durch eine komponentenbasierte Softwarearchitektur ist es dem Entwickler möglich, die computergrafische Bildsynthese durch die

Implementierung von Visualisierungsbausteinen fortwährend zu erweitern beziehungsweise zu modifizieren. Ferner wird dem Anwender gewährt, die jeweiligen Manipulationsmöglichkeiten über eine grafische Benutzerschnittstelle im interaktiven Prozess einzusetzen. Die Komponenten sind für eine möglichst flexibel anpassbare und skalierbare Arbeitsumgebung als geschichtete Komponentenarchitektur konzipiert und umgesetzt. Die Visualisierungsbausteine können entsprechend ihrer Funktionalität in der Arbeitsumgebung in drei Ebenen untergliedert werden:

- *Atomare Komponenten* bieten grundlegende Abbildungsverfahren, die den Kern einer performanten und effizienten Bildsynthese bilden und ferner spezifizieren. Die Komponenten ermöglichen eine flexible Parametrisierbarkeit des Bildsyntheseprozesses, um eine Grundlage für die funktionalen und die operativen Komponenten bereitzustellen.
- *Funktionale Komponenten* bilden eine konzeptionelle Mittelschicht in der Arbeitsumgebung. Diese nutzen die Dienste atomarer Komponenten und bilden, durch Erweiterung und kombinierte Nutzung ihrer Funktionalität, Softwarebausteine für nichtlineare und multiprojektive Darstellungstechniken.
- *Operative Komponenten* werden durch die Kombination von funktionalen Komponenten gebildet. Die Elemente dieser Ebene bieten einen Funktionsumfang für die parametrisierte Ausgestaltung sowie Kopplung von funktionalen Visualisierungsbausteinen zur Gestaltung einer konkreten 3D-Benutzerschnittstelle und der Umsetzung eines interaktiven Nutzer-Bild-Dialogs.

Die Verwirklichung einer flexiblen Arbeitsumgebung als komponentenbasierte Softwareplattform bedingt eine Reduzierung von Abhängigkeiten zwischen den Visualisierungsbausteinen. Gleichzeitig müssen die Komponenten im Anwendungskontext miteinander kollaborieren, sodass nach der Zerlegung des Funktionsumfanges eine funktionale und situative Zusammenführung erforderlich ist. Dies wird durch die Bereitstellung der Visualisierungsbausteine als Bundles sowie deren Einbindung in eine Service-Struktur gewährleistet, die durch die POCO Open Service Plattform (OSP) in die Arbeitsumgebung integriert ist. OSP bietet eine Service-Registrierung, bei der sich ein Bundle-Service eintragen kann. Die Funktionalität der Bundles wird durch den Service bereitgestellt und mittels eines Dienstverzeichnisses lokal veröffentlicht, sodass andere Bundles die Funktionalität nutzen können. Durch das Service-Model wird zum einen die konkrete Implementierung von der Verwendung der Dienste abstrahiert, infolgedessen die Anbieter von Services austauschbar sind. Zum anderen ermöglicht die indirekte Kopplung, dass Dienste genutzt werden können, ohne dass ein dienstnutzendes Bundle an dessen Bereitstellung beteiligt ist. Folglich ist es unabhängig von einem konkreten Anbieter. Zur Laufzeit des Bildsprache LiveLab werden die Visualisierungsbausteine in der *Laufzeitumgebung*, der POCO Open Service Plattform, auf Basis des Komponenten-Lebenszyklus ausgeführt (siehe Unterabschnitt 4.2.1) und bilden in der Folge ein dynamisches, serviceorientiertes Komponentensystem.



### 4.3.2 Aufbau und Funktionsweise der Softwarearchitektur

Die Komponenten des Bildsprache LiveLab sind entsprechend ihrer Funktionalität in die *atomare Ebene*, die *funktionale Ebene* oder die *operative Ebene* eingeordnet. Darüber hinaus sind diese, wie es in Abbildung 23 als Komponentenmodell aufgezeigt ist, über Dienste miteinander verbunden. Die atomare Ebene besteht aus zwei Komponenten: *BillCore* und *WxRunner*. Die zwei Bundles bieten grundlegende Funktionen für die Entwicklung und Nutzung von funktionalen Komponenten durch den Zugriff auf das GUI-Toolkit *wxWidgets* einerseits und die Grafik-Engine *OpenSceneGraph* andererseits. Die Komponente *WxRunner*, die keine Abhängigkeiten zu anderen Komponenten aufweist, ermöglicht zum einen die Erstellung von Fenstern für traditionelle 2D-Benutzerschnittstellen, zum anderen von Programmfenstern für die Darbietung von 3D-Szenen. Die Bereitstellung der Funktionalität dieser Komponente erfolgt über einen Dienst, den *WindowManagerService*. *BillCore* ist die zweite Komponente in der atomaren Ebene. Dieser Softwarebaustein stellt über den Dienst *BillService* eine Schnittstelle zur Grafik-Engine zur Verfügung. Durch den Service sind atomare Funktionen der Bildsynthese in der Arbeitsumgebung verfügbar. Beispielsweise stellt der Dienst den Zugriff auf die computergrafische Kamera und auf Szenendaten für eine Gestaltung des Bildraums sicher. Infolgedessen können mithilfe der Schnittstelle zu *OpenSceneGraph* die Modifikationen am Abbildungsprozess und an den Elementen des Objektraums durch funktionale Komponenten der BiLL-Anwendung vollzogen werden. Dadurch können ein einzelnes Abbild oder mehrere 3D-Visualisierungen verbildlicht werden.

Die funktionale Ebene bilden einerseits die Komponenten *WxEditor* und *WxBundleAdmin*, andererseits die Visualisierungsbausteine zur Manipulation der computergrafischen Kamera (siehe Unterabschnitt 3.3.2). Die Softwarekomponenten stellen lineare und nichtlineare Darstellungsverfahren bereit, die nicht durch den konventionellen Projektionsvorgang – basierend auf Komponenten der atomaren Ebene – abgebildet werden. Exemplarisch sind in Abbildung 23 die Komponenten zur Erzeugung multiperspektivischer und krummliniger Darstellungen in das Komponentenmodell eingefügt. Die Verwaltung zur Integration und Ausführung der Visualisierungsbausteine erfolgt durch die *WxBundleAdmin*-Komponente, die den Lebenszyklus von Bundles basierend auf der OSGi-Spezifikation steuert. Ein Überblick über die durch den *WxBundleAdmin* verwalteten Visualisierungskomponenten ist in Abbildung 24 anhand der verschiedenen Visualisierungsformen aufgezeigt. Das *WxEditor*-Bundle bietet ferner die Möglichkeit, die Benutzeroberfläche des Bildsprache LiveLab um verfahrensrelevante 2D-Interfaceelemente zu erweitern. Der Zugriff auf die Elemente für die grafische Benutzeroberfläche erfolgt durch den *WindowManagerService* der *WxRunner*-Komponente. Dadurch können dem Anwender Interaktionsmöglichkeiten zur Initiierung, Parametrisierung und Modifizierung von Visualisierungsverfahren für die interaktive Gestaltung des Bildraums zur Laufzeit der Arbeitsumgebung bereitgestellt werden.

Die Komposition von Bundles, welche im Zusammenwirken die BiLL-Anwendung bilden, erfolgt durch den Einsatz der operativen Komponenten *Bill4Desktop* und *InteractivePanels*. Das *Bill4Desktop* Bundle spezifiziert hierbei das konkrete Erscheinungsbild des Bildsprache LiveLab. Die Komponente instanziiert ein Editor-Fenster unter Verwendung der *WxEditor* Komponente. Das Editor-Fenster ermöglicht das Laden und Speichern von 3D-Szenen



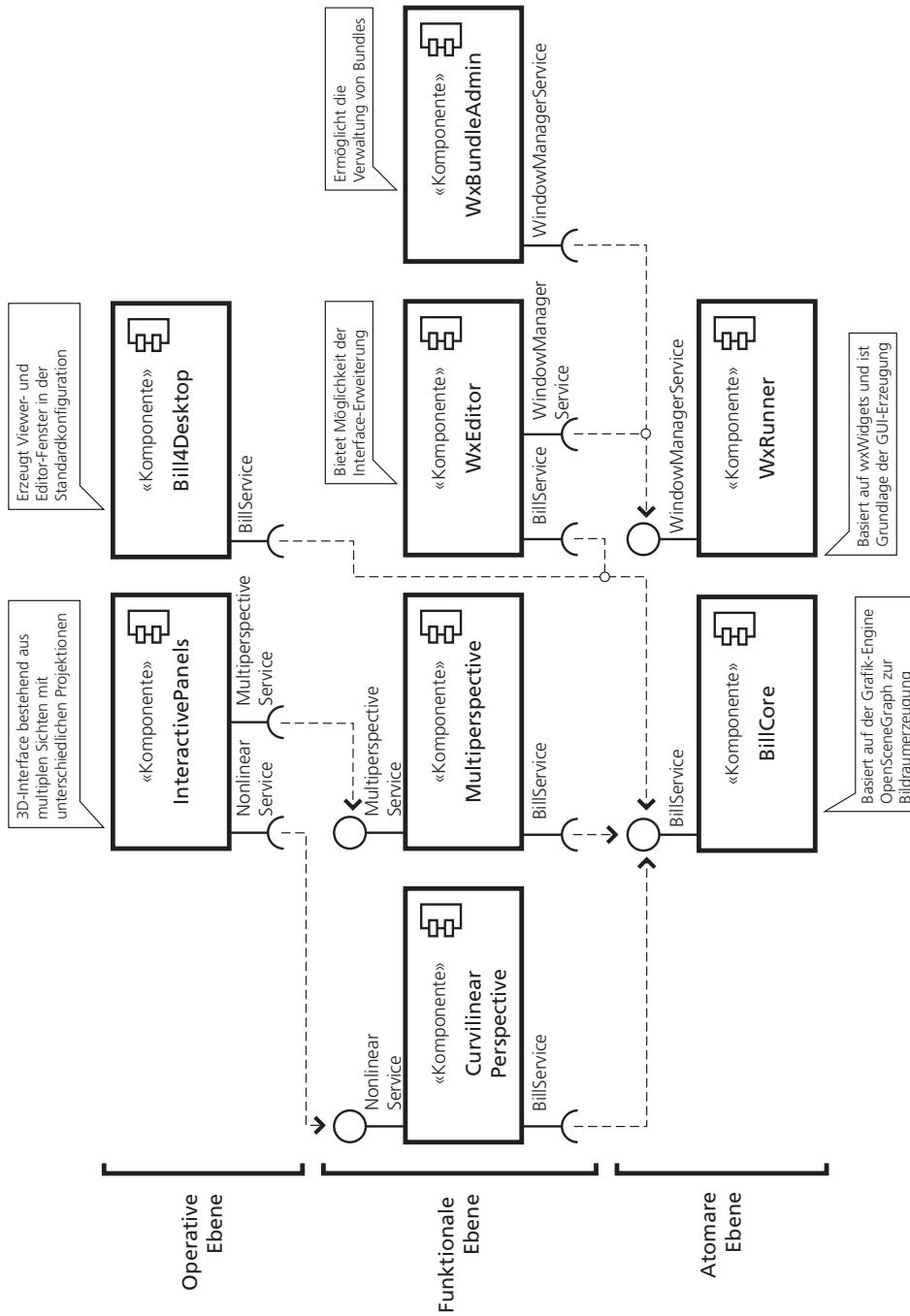


Abbildung 23: Softwarearchitektur des Bildsprache LiveLab als Komponentenmodell zeigt wesentliche Komponenten des Systems auf drei Ebenen, die durch Dienste miteinander verbunden sind.

und infolgedessen die Verfügbarkeit geometrischer Daten in BiLL. Weiterhin wird die Darstellung der aktuell geladenen Szene in Form eines Szenengraphs innerhalb des Editor-Fensters präsentiert. Auf Basis der Funktionalität des WxEditor-Bundles kann darüber hinaus der strukturelle Aufbau der Szene modifiziert werden, indem der Szenengraph um Knoten erweitert wird oder Elemente aus diesem gelöscht werden. Ferner bietet die Komponente die Möglichkeit, den Editor um Benutzerschnittstellenelemente zu erweitern. Die Erweiterung erfolgt, sobald Visualisierungskomponenten während der Laufzeit hinzugefügt werden, woraufhin die neuen Funktionen durch grafische Benutzerschnittstellenelemente im Editor repräsentiert werden. Für die Verbildlichung der 3D-Benutzerschnittstelle, unter Berücksichtigung der Interaktionen des Nutzers im Editor, spezifiziert die Komponente BiLL4Desktop ferner ein Viewer-Fenster als Darstellungsfläche für die Visualisierung einer geladenen 3D-Szene. Dies erfolgt mithilfe des BillServices der BillCore Komponente. Damit legt die Komponente das Fundament für die Darstellung von interaktiven Bildräumen. Die linearen und nichtlinearen Visualisierungen, die durch die Bundles der funktionalen Ebene bereitstehen, können nunmehr durch die Komponente InteractivePanels auf der Darstellungsfläche des Viewer-Fensters verortet werden. Dabei können, aufgrund der bereitgestellten Dienste der funktionalen Komponenten, neue Sichten erstellt und bestehende Sichten modifiziert werden. Die interaktiven Bildräume können durch die InteractivePanels-Komponente ferner miteinander verknüpfen werden, um koordinierte multiple Sichten zu verwirklichen. Das entstehende konkrete Erscheinungsbild des Bildsprache LiveLab mit der grafischen Ausgestaltung des Viewer- und des Editor-Fensters sowie dessen Nutzung aus Anwendersicht wird weiterführend in Abschnitt 6.1 beschrieben.

### 4.3.3 Komponentenkombination zur Bildraummodifikation

Auf Basis der Komponentenarchitektur des Bildsprache LiveLab entstand eine Vielzahl von Visualisierungsbausteinen. Einen Schwerpunkt der Entwicklungen bilden die im Abschnitt 3.3.3 systematisierten Projektionsverfahren und deren Anwendung im Kontext konkreter Visualisierungsziele (siehe Unterabschnitt 3.5.1). Hierzu wurden Visualisierungsverfahren als unabhängige und gleichzeitig kombinierbare Komponenten in BiLL verwirklicht. Die Abbildung 24 zeigt die verschiedenen Visualisierungsformen, die in das Bildsprache LiveLab integriert sind, und ordnet diese in die Komponentenebenen der BiLL-Anwendung ein. Die Abbildung zeigt, dass lineare Projektionsverfahren auf der atomaren Ebene in die Experimentierungsumgebung integriert sind, indem diese durch die Grafik-Engine bereitgestellt werden (siehe Abbildung 24-A,-B). Die Abbildungsverfahren sind über die Dienste der BillCore-Komponente in der Arbeitsumgebung verfügbar. Zusätzlich zu den Kameraparametern, die üblicherweise in 3D-Anwendungen spezifizierbar sind, beispielsweise die Position und die Ausrichtung, ist durch den Dienst des BillCore-Bundles die Verlagerung des Hauptpunktes mittels einer Scherung des Kamerasichtkörpers, in das Bildsprache LiveLab integriert (siehe 3.3.2.5, Abbildung 24-C und weiterführend [EBNER 2007, S. 88 ff.]).

Auf Basis der atomaren Dienste bieten funktionale Visualisierungskomponenten Darstellungstechniken, die krummlinige, multiperspektivische und invers-zentralprojektive Bilder erzeugen können. Die Integration krummliniger Perspektiven in eine 3D-Benutzerschnittstelle erfolgt

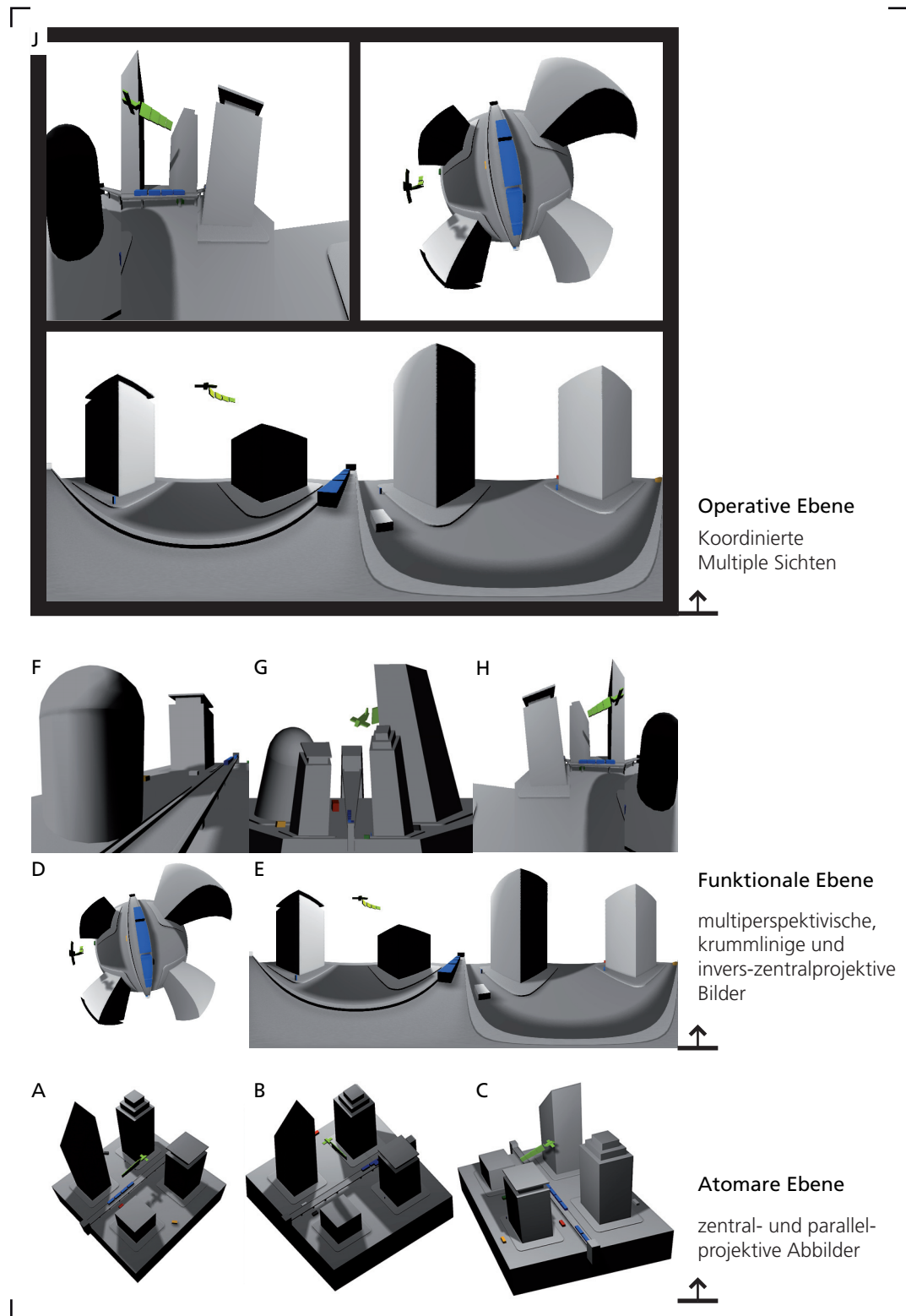


Abbildung 24: Darstellungsformen im Bildsprache LiveLab, die als Bundles integriert sind, verdeutlichen den Aufbau der BiLL-Architektur. Auf Basis von atomaren Komponenten (A-C) ermöglichen Bundles der funktionalen Ebene die Erzeugung nichtlinearer Darstellungen (D-H), die in der operativen Ebene in koordinierten multiplen Sichten (J) montiert werden können (Quelle: BiLL [@AG-TV]).

dabei auf Basis einer Komponente zur Erzeugung kurvilinearere Projektionen. Die Erzeugung des Bildraums findet mithilfe von mathematischen Abbildungsfunktionen statt, die ebenso in geodätischen Kartenprojektionen eingesetzt werden (vgl. [SNYDER & VOXLAND 1989, S. 5 ff.]). Die verfahrenstechnische Umsetzung erfolgt hierbei in zwei Schritten: Zunächst wird die visuelle Struktur auf die Innenseite einer abstrahierten Kugel projiziert, um die dreidimensionale Szene umfassend und gleichzeitig verzerrungsfrei abzubilden. Im zweiten Schritt wird das Abbild der Projektionsfläche auf eine planare Bildebene überführt (siehe 3.3.2.2 und weiterführend [SCHMIDT 2009, S. 74 ff.]). Die bereitgestellten Abbildungsfunktionen ermöglichen die Erzeugung von Fischaugen- (siehe Abbildung 24-D) und Panoramadarstellungen (siehe Abbildung 24-E), wobei die verwendete Abbildungsfunktion und der horizontale beziehungsweise vertikale Sichtbereich über die Benutzeroberfläche des BiLL-Editors festgelegt werden können. Weiterhin ist in der Arbeitsumgebung die Erzeugung multiprojektiver Darstellung möglich. Die Umsetzung einer objektraumbasierten Multiperspektive wird in BiLL durch zwei Bundles – ein objekt- und ein kamerabasiertes Verfahren – gewährleistet. Das Bundle für einen objektbasierten Ansatz verändert die Geometrie eines Objektes in Abhängigkeit zur Kameraposition, um dieses in einer separaten Perspektive zu visualisieren (siehe 3.3.2.1, Abbildung 24-F und weiterführend [WOJDZIAK 2007, S. 65 ff.]). Der kamerabasierte Ansatz erfolgt unter Einsatz von Helferkameras, bei dem einzelne Objekte einer Szene abgebildet und in einer Bildkomposition zu einem Gesamtbild auf der Bildebene zusammengeführt werden (siehe 3.3.2.3 und weiterführend [WUTTIG 2008, S. 26 ff.]). Beide Verfahren können parallel als Gestaltungswerkzeuge eingesetzt werden, wobei die Objekte, die in einer separaten Perspektive dargestellt werden sollen, über das grafische Interface des Bildsprache LiveLab manuell selektiert werden. Eine bildraumbasierte Multiperspektive wird in BiLL durch die Aufteilung der Bildebene in vertikale Streifen umgesetzt, wobei jedem Streifen eine unabhängige Kamera zugeordnet wird (siehe 3.3.2.3, Abbildung 24-H und weiterführend [MÜNCH 2008, S. 73 ff.]). Die Anzahl der Kameras und der zu visualisierende Sichtbereich können durch den Anwender festgelegt werden, während verschiedene Kameragrundanordnungen über die grafische Benutzeroberfläche des Bundles angeboten werden. Ein weiterer Visualisierungsbaustein bietet die Verwirklichung einer umgekehrten Perspektive mithilfe eines invers-zentralprojektiven Abbildungsvorgangs (siehe 3.3.2.5 und weiterführend [WALTHER 2012, S. 99 ff.]). Aufbauend auf dem computergrafischen Kamerasichtkörper wird auf der funktionalen Ebene die vordere und die hintere Sichtbegrenzungsebene ausgetauscht, um diese Form linearer Abbilder darzustellen (siehe Abbildung 24-G).

Die Zusammenführung von Darstellungen, die auf Basis von atomaren und funktionalen Diensten erzeugt werden, erfolgt über die Komponente zur Koordination multipler Sichten. Diese ermöglicht die gleichzeitige Darstellung mehrerer Sichten auf einen Objektraum. Dazu werden mehrere Abbildungen einer visuellen Struktur erzeugt (siehe 3.3.2.4, Abbildung 24-J und weiterführend [WOJDZIAK U. A. 2011c; STARKE U. A. 2011]). Dem Anwender wird für die Gestaltung einer 3D-Benutzerschnittstelle die Möglichkeit gegeben, die Anzahl der Sichten, deren Position sowie deren Größe auf dem Darstellungsbereich festzulegen und die Bildräume auf Basis eines Regelwerkes miteinander zu koppeln.

### 4.3.4 Entwicklungsstufen der Arbeitsumgebung

Aufbauend auf den Betrachtungen zur Architektur des Bildsprache LiveLab und den entwickelten Visualisierungskomponenten werden nun die verschiedenen Stufen der Entwicklung und Nutzung der Arbeitsumgebung erläutert. Die Stufen im Entwicklungsprozess des Bildsprache LiveLab sind mit *Komponente*, *Komposition* und *Anwendung* bezeichnet und beschreiben die Arbeitsschritte die für den Einsatz von Visualisierungskomponenten in der interaktiven Arbeitsumgebung notwendig sind.

Die Stufe der Komponentenentwicklung umfasst die Implementierung von atomaren, funktionalen und operativen Softwarekomponenten sowie die Spezifikation von Diensten für die Bereitstellung der Funktionalität in der Arbeitsumgebung. Weiterhin erfolgen der Entwurf und die Implementierung einer Benutzeroberfläche zur nutzerseitigen Ausführung und Modifikation der Bundle-Funktionalität im Bildsprache LiveLab. In der Regel sind zur Erfüllung der Aufgaben innerhalb der Komponentenentwicklung Computergrafiker, Softwareentwickler, aber auch Interface- und Interaktionsdesigner beteiligt. Die Entwicklungen haben das Ziel, Verfahrenstechniken zur Erzeugung von computergrafischen Bildräumen für 3D-Benutzerschnittstellen als algorithmische Beschreibung in die bestehende Arbeitsumgebung zu integrieren. Bei der Implementierung der Bundles kann durch den BillService auf die zugrundeliegende Grafik-Engine und durch den WindowManagerService auf das GUI-Toolkit, zugegriffen werden. Hierdurch stehen dem Entwickler bei der Bundle-Implementierung Methoden zum Szenengraphmanagement und zur Abbilderzeugung sowie zur Erstellung von Benutzeroberflächen zur Verfügung. Bei der Implementierung der Komponenten kommen Werkzeuge im Bereich der *integrierten Entwicklungsumgebungen (IDE)* zum Einsatz. Die Implementierung der Dienste erfolgt ebenfalls durch die Erstellung von Implementierungsdateien mithilfe von Entwicklungsumgebungen und Quellcodekompilern.

Die Kompositionsstufe umfasst die Kombination und Konfiguration von bestehenden Visualisierungskomponenten für einen Anwendungskontext. Die Konfiguration der Arbeitsumgebung wird mithilfe des WxBundleAdmins vollzogen. Dazu wird die Grundkonfiguration der BiLL-Anwendung genutzt, wie diese in der Bill4Desktop Komponente implementiert ist. Von dieser Grundkonfiguration ausgehend – bestehend aus dem Editor-Fenster und einem Viewer-Fenster – werden Darstellungstechniken durch das Installieren und das Starten von Bundles über den Bundle-Admin, ergänzt. Die gestarteten Bundles können in einem nachfolgenden Schritt über die Bedienoberflächen, die in der Komponentenstufe erstellt wurden, im Editor-Fenster konfiguriert werden. Die Konfiguration der Arbeitsumgebung kann durch Experten verschiedener Fachgebiete und ohne Programmierkenntnisse vorgenommen werden. Dazu wird zunächst die szenengraphbasierte Beschreibung des Objektraums, die mit Autorenwerkzeugen oder Werkzeugen zur Datenrepräsentation erzeugt wird, in die BiLL-Anwendung geladen. Mithilfe der Visualisierungsbausteine kann nunmehr neben der szenengraphbasierten Modifikation des Objektraums die Veränderung von Bildräumen, die Erzeugung mehrerer Sichten sowie deren Kopplung durchgeführt werden. Das Resultat der Kompositionsstufe ist eine konfigurierte 3D-Benutzerschnittstelle unter Einbeziehung von konkreten Daten beziehungsweise visuellen Strukturen. Die konkrete Ausgestaltung der 3D-Benutzerschnittstellen, auf Basis der komponierten und konfigurierten Bundles, wird in einer Konfigurationsbeschreibung festgehalten. In der Folge steht diese für

den Einsatz des Bildsprache LiveLab im konkreten Anwendungskontext zur Verfügung. Eine Ausnahme der programmierfreien Komposition und Konfiguration stellt die Verwirklichung einer spezifischen Arbeitsumgebung auf Basis einer veränderten Anwendungsgrundkonfiguration dar. Die Konfiguration für das Bildsprache LiveLab als eine Desktopanwendung ist durch die Komponente *Bill4Desktop* sichergestellt. Die Realisierung von Ausprägungen wie beispielsweise *Bill4CAVE* oder *Bill4Tablet* muss zunächst durch eine Implementierung von entsprechenden Komponenten dem Bildsprache LiveLab hinzugefügt werden, bevor eine Komposition und Konfiguration der Arbeitsumgebung für diese Umgebungen stattfinden kann.

Die Anwendungsstufe umfasst die Nutzung des konfigurierten Bildsprache LiveLab im Anwendungs- und Forschungskontext. Das interaktive System wird in dieser Stufe vorwiegend von Interface- und Interaktionsdesignern genutzt, welche die Darstellungstechniken im Hinblick auf deren Nutzung zur Verwirklichung von Visualisierungszielen untersuchen und erproben. Ferner wird die Arbeitsumgebung von Wissenschaftlern aus dem Bereich der Wahrnehmungspsychologie und der kognitiven Ergonomie genutzt, indem diese anwendungsbezogene Überprüfungen von Darstellungsverfahren durchführen und das Bildsprache LiveLab als Werkzeug zur experimentellen Gestaltung von 3D-Benutzerschnittstellen interaktiver Systeme einsetzen.

## 4.4 Zusammenfassung und Diskussion

In diesem Kapitel konnte aufgezeigt werden, dass die Visualisierungsplattform Bildsprache LiveLab eine Umsetzung von Darstellungsverfahren, wie diese im dritten Kapitel vorgestellt werden, als Visualisierungsbausteine bietet. Dies wird durch den Einsatz von POCO Open Service Plattform als Laufzeitumgebung, von OpenSceneGraph als Grafik-Engine zur Visualisierung von 3D-Szenen und von wxWidgets als GUI-Toolkit für den interaktiven Prozess mit dem Anwender erreicht. Die Architektur und die verschiedenen Entwicklungsstufen des Bildsprache LiveLab zeigen, dass auf Basis der verwendeten Technologien die Anforderungen, die an die Arbeits- und Experimentierumgebung gestellt wurden, zum Zeitpunkt der Arbeit mit wenigen Einschränkungen erfüllt sind. Eine bisher unerfüllte Anforderung ist die Adaptierbarkeit von Benutzerschnittstellen für verschiedene Personengruppen im Bildsprache LiveLab. BiLL ist gegenwärtig als Expertensystem für Anwender aus dem Bereich der Technischen Visualistik konzipiert und umgesetzt.

In der Arbeitsumgebung wurden bisher eine Vielzahl von Darstellungstechniken, jedoch nicht alle Ausprägungen der Visualisierungsverfahren – wie diese im Abschnitt 3.5 vorgestellt werden – umgesetzt. Neben deren Beschreibung und Konzeption ist die häufig komplexe Realisierung der Funktionalität zu leisten. Dies käme einer Re-Implementierung einer großen Anzahl aus der Literatur bekannten Visualisierungsverfahren gleich. Mit der Umsetzung von ausgewählten Darstellungsformen, basierend auf der Systematisierung in Abschnitt 3.3, erfolgt vielmehr die gezielte Bereitstellung eines Grundfundus an kombinierbaren Darstellungsverfahren, mit dem eine Verwirklichung von Visualisierungszielen (siehe Abschnitt 3.5) erreichen werden kann. Die Visualisierungsbausteine bieten keine Verfahrensweise, die eine

vollautomatische Überführung eines Objektraums auf das Ausgabegerät verwirklicht. Ein experimenteller Charakter der Arbeitsumgebung ist gegeben und wird in der Entwicklung der Bildsprache LiveLab forciert, da die Forschungen auf dem Gebiet des Non-Photorealistic Rendering noch nicht den Stand erreicht haben, der den zielführenden Einsatz von Visualisierungsverfahren ohne die Bereitstellung von Freiheitsgraden möglich werden lässt. Eine solche Einschränkung würde, mit dem Fokus auf die Wiederverwendung von Visualisierungskomponenten, zu einer Verminderung der Flexibilität führen, die das kreative Entwickeln von neuen Ansätzen und Verfahrensweisen verhindert. Ferner würde der Einsatz einheitlicher Darstellungsverfahren die Anpassung an spezifische Visualisierungsziele einschränken. Gleichzeitig ist mit dem Bildsprache LiveLab eine Basis für künftige Weiterentwicklungen im Bereich der 3D-Interfaceentwicklung gelegt. Die komponentenbasierte Umsetzung von Darstellungsverfahren ermöglicht einerseits die Nutzung von Visualisierungsverfahren in einer interaktiven Umgebung und begünstigt andererseits durch den Komponentencharakter deren Portierung in andere 3D-Applikationen.



## 5 MosaIK: Modellbasierte 3D-Interfacekomposition

Im Verlauf der Entwicklung von Benutzerschnittstellen sind die grundlegenden Anforderungen an ein grafisches Interface, trotz technologischer Verbesserungen und grafischer Weiterentwicklungen, unverändert geblieben. Eine Mensch-Computer-Schnittstelle muss einerseits die Eingaben eines Anwenders aufnehmen und interpretieren, andererseits die resultierenden Ausgaben darbieten. Dieser Vorgang wird im dritten Kapitel als interaktiver Prozess zwischen Mensch und Computer verdeutlicht und mit dem Fokus auf 3D-Benutzerschnittstellen erläutert. Ferner wird dieser Kommunikationsprozess im Hinblick auf mögliche Visualisierungsziele analysiert und systematisiert. Die Bereitstellung interaktionsförderlicher Bildstrukturen als Softwarebausteine für eine interaktive Anwendung zeigt im vierten Kapitel weiterführend die praktische Umsetzbarkeit eines Nutzer-Bild-Dialoges in einer interaktiven 3D-Echtzeitumgebung. Der gezielte Einsatz von Darstellungstechniken für eine bildgestützte Interaktion in 3D-Anwendungen wird nunmehr in ein modellgestütztes Vorgehen überführt, um deren generischen Einsatz in einem methodischen Entwurfsprozess von Benutzerschnittstellen zu erwirken. Dem Vorgehen wird hierzu ein Phasenkonzept zugrunde gelegt. Mit einem Durchlaufen der Phasen wird der Entwurfsprozess von 3D-Interfaces geführt, der mit einer Anforderungsspezifikation beginnt und in einer plattformspezifischen Beschreibung der Benutzerschnittstelle mündet. In den einzelnen Phasen werden, entsprechend des zunehmenden Konkretisierungsgrades des Interfacedesigns, Modelle und Methoden zur Beschreibung von Benutzerschnittstellen bereitgestellt. Als Resultat werden hierdurch methodische Hilfsmittel geboten, mittels derer ein Entwurfsprozess vollzogen werden kann, der wie von GROH gefordert, eine grafische Benutzerschnittstelle als eine „interaktive und gestalterische Ganzheit“ [GROH 2007, S. 111] spezifizierbar macht.

In diesem Kapitel werden zunächst die Herausforderungen im 3D-Interfaceentwurf aufgezeigt und system- sowie nutzerbasierte Vorgehensweisen für einen Umsetzungsprozess beleuchtet (5.1). Anschließend wird die Verfahrensweise für eine »modellbasierte 3D-Interfacekomposition (MosaIK)« eingeführt (5.3). Dazu werden zunächst die eingesetzten Sprachen und Notationen erläutert, welche die notwendige Präzision im Entwurfsprozess sicherstellen (5.2). Ferner werden für die Beschreibbarkeit von 3D-Interfaces UML-Profile als Erweiterung der Modellierungssprache UML (5.5) und eine XML-basierte Auszeichnungssprache (5.6) eingeführt. Da jedoch durch das modellgestützte Vorgehen die Erfüllung softwareergonomischer Anforderungen nicht sichergestellt wird, werden vorgehensbegleitend Entwurfsmuster für die Etablierung, die Anwendung und die Wiederverwendung von Entwurfswissen für eine effektive 3D-Interfaceerstellung in die MosaIK-Verfahrensweise integriert.

## 5.1 Entwurfsverfahren und deren Eignung für 3D-Benutzerschnittstellen

### 5.1.1 Herausforderungen im Entwurfsprozess

Die Erzeugung grafischer Benutzerschnittstellen für einen konkreten Anwendungskontext setzt ein umfangreiches Expertenwissen voraus. In [FUJISHIRO U. A. 1997, S. 245] wird sogar der Begriff „*visineer*“ verwendet, mit dem FUJISHIRO die Entwicklung von Visualisierungen und den Entwurf grafischer Benutzerschnittstellen als eine Ingenieursdisziplin beschreibt. Experten auf den Gebieten des Interface- und Interaktionsdesigns sind bei der Entwicklung von Benutzerschnittstellen mit der Herausforderung konfrontiert, den Anwendern interaktiver Systeme die Bewältigung von Aufgaben und die Präsentation von Ergebnissen zu ermöglichen (vgl. [SCHUMANN & MÜLLER 2000, S. 5–6]). Die Verwendung interaktiver Systeme durch oftmals heterogene Nutzergruppen und deren Anforderungen an das grafische Interface hat dazu beigetragen, dass der menschliche Faktor im Interfacedesign zunehmend Gegenstand der Forschung geworden ist (vgl. [TORY & MÖLLER 2004]). In diesem Zusammenhang motiviert SHNEIDERMAN den Entwurfsprozess von Interfaces ausgehend vom Anwender und von dessen Aufgaben zu vollziehen: *“When an interactive system is well designed, the interface almost disappears, enabling users to concentrate on their work, exploration or pleasure”*<sup>58</sup> [SHNEIDERMAN 2004, S. 12].

Eine der zentralen Aufgaben bei der Verwirklichung eines Nutzer-Bild-Dialoges ist es, die bildgestützten Interaktionsabläufe in einer Form zu entwerfen, dass der Anwender situativ und zielgerichtet mit dem System kommunizieren kann. Bei der Umsetzung visueller Repräsentationen und Dialogtechniken sind dabei vielseitige Einflussfaktoren zu berücksichtigen, die keine ganzheitliche Lösung für die Realisierung einer Benutzerschnittstelle zulassen. In der Folge dienen Kriterien dazu, den Gestaltungsprozess zu leiten und Ergebnisse bewertbar und objektivierbar zu machen. So fordert HERCZEG eine anwendungsgerechte Gestaltung von Benutzerschnittstellen unter der Berücksichtigung von Kriterien der Gebrauchstauglichkeit und der menschlichen visuellen Wahrnehmung (vgl. [HERCZEG 2006, S. 5]). Ferner identifizieren SCHUMANN und MÜLLER die Angemessenheit als wichtiges Merkmal zur Bewertung visueller Repräsentationen (vgl. [SCHUMANN & MÜLLER 2000, S. 12]). PREIM und DACHSELT betonen, dass bei der Gestaltung eines Nutzer-Bild-Dialoges statische und dynamische Aspekte zu berücksichtigen sind (vgl. [PREIM & DACHSELT 2010, S. 413]). Dahin gehend konkretisiert GROH den Gestaltungsprozess interaktiver Systeme durch die Einführung *syntaktischer Felder* unter Berücksichtigung statischer und dynamischer Schnittstellencharakteristiken (vgl. [GROH 2007, S. 99 ff.]) und verknüpft diese für einen methodischen Interfaceentwurf durch *Übergangssyntagmen*. In Bezug auf 3D-Interfaces weißt GREEN darauf hin, dass Gestaltungskriterien konventioneller 2D-Benutzeroberflächen sehr wohl gültig sind. Die 3D-Interfaces verfügen jedoch über weitere Anforderungen, unter Berücksichtigung spezifischer Charakteristiken (vgl. [GREEN & JACOB 1991, S. 229–230]), die traditionelle Benutzeroberflächen wiederum nicht besitzen (vgl. [JACOB U. A. 1999, S. 3–4]).

<sup>58</sup> „Wenn eine interaktive Anwendung gut gestaltet ist, tritt die Benutzerschnittstelle in den Hintergrund und befähigt den Anwender sich auf seine Aufgabe, die Erforschung oder das Vergnügen, zu konzentrieren.“

Es werden nunmehr wesentliche Probleme und resultierende Herausforderungen im Entwurfsprozess von 3D-Benutzerschnittstellen, basierend auf den Erkenntnissen wissenschaftlicher Betrachtungen und derzeitiger Anwendungen, dargelegt:

- Die Entwicklung von 3D-Benutzerschnittstellen für interaktive Anwendungen ist im Schnittgebiet von Disziplinen angesiedelt, in dem Interfacedesigner mit Softwareentwicklern und Experten der jeweiligen Anwendungsdomäne in einem interdisziplinären Team zusammenwirken. Oftmals besteht jedoch zwischen den Beteiligten ein Kommunikations- und Verständnisdefizit aufgrund unterschiedlicher Sprachstile und Abstraktionsniveaus bei der Beschreibung der Anforderungen an ein 3D-Interface. Eine aus diesem Umstand resultierende, defizitäre Spezifikation kann zu Fehlentscheidungen und zu einem teilweisen oder vollständigen Neuentwurf von Benutzerschnittstellen interaktiver Systeme führen.
- Die Implementierung von Benutzerschnittstellen ist ein wesentlicher Teil des gesamten Softwareentwicklungsprozesses. Die zugrunde liegende Spezifikation eines 3D-Interface kann als Teil der Anforderungsspezifikation sowie der Systemspezifikation verstanden werden. In welcher Art und Weise jedoch die Spezifikation einer Anwendungsentwicklung zu strukturieren und zu beschreiben ist, damit eine 3D-Benutzerschnittstelle anforderungskonform umgesetzt werden kann, bleibt von den Richtlinien und Modellen der Softwareergonomie weitgehend unbeantwortet.
- Es existieren allgemeine Hinweise und Hilfestellungen, welche die Entwicklung von 3D-Benutzerschnittstellen unterstützen. Diese sind für ein grundsätzliches Verständnis wesentlich und bieten Einblicke in elementare Aspekte der Entwicklung von 3D-Interfaces, sind jedoch durch eine fehlende Eindeutigkeit nicht konkret in deren Anwendung. In der Folge ist eine große Anzahl an Iterationsschritten bei einer Umsetzung notwendig, wodurch sich die Entwicklungszeit von interaktiven 3D-Anwendungen verlängert.
- Im Anwendungsumfeld verfügen interaktive Systeme bisweilen über innovative Benutzerschnittstellen für einen effizienten Nutzer-Bild-Dialog. Die Entwurfsentscheidungen und die resultierenden Spezifikationen, die zu diesen 3D-Anwendungen führten sind oftmals nicht mehr existent, nicht einsehbar oder schlecht dokumentiert. Es bieten sich daher kaum Möglichkeiten, das Wissen über die Gestaltung und Umsetzung effizienter Benutzerschnittstellen zu extrahieren und in der Folge in zukünftigen Interfaceentwicklungen wiederzuverwenden.
- Eine Realisierung interaktiver Systeme ist häufig mit einer vollständigen Neuentwicklung der 3D-Benutzerschnittstelle verbunden. Obwohl existierende Anwendungen oftmals vergleichbare Anforderungen besitzen, erfolgt, aufgrund fehlender Beschreibungsmöglichkeiten, keine Entwicklung von plattform- und anwendungsunabhängigen 3D-Benutzerschnittstellen. In der Folge wird keine Senkung des Implementierungsaufwandes zukünftiger 3D-Anwendungen erreicht.

Die identifizierten Probleme bekräftigen die Motivation für einen gestützten Entwurfsprozess von interaktiven 3D-Benutzerschnittstellen. Dafür müssen geeignete Vorgehensweisen und

zweckmäßige Beschreibungsformen bereitgestellt werden, welche durch eine heterogene Nutzergruppe in den Phasen der Benutzerschnittstellenentwicklung übergreifend verstanden genutzt werden können.

### 5.1.2 System- und benutzerzentrierte Vorgehensweisen

Die Ergebnisse einer Analyse von MYERS und ROSSON aus dem Jahr 1992 verdeutlichen, dass die Interfaceentwicklung einen zeit- und kostenintensiven Anteil in der Umsetzung einer interaktiven Anwendungen einnimmt (vgl. [MYERS & ROSSON 1992, S. 199]). Dieser Entwicklungsaufwand ist notwendig, da die Benutzerschnittstelle für die Bedienbarkeit, die Akzeptanz und letztendlich für den Erfolg einer Anwendung ausschlaggebend ist. Aufgrund der gestiegenen Anforderungen einerseits und einer stetig wachsenden Anzahl heterogener Ausgabegeräte andererseits ist auch in den aktuellen Entwicklungen der Realisierungsaufwand für grafische Benutzerschnittstellen beträchtlich und wird von PETRASCH in [PETRASCH 2007, S. 5] auf mindestens 50% der Gesamtentwicklungszeit beziffert. Die Ursache liegt darin begründet, dass ein grafisches Interface in aktuellen Softwareprojekten nicht nur für eine spezifische Applikation, sondern für ein Softwaresystem mit unterschiedlichen Konfigurationen und für unterschiedliche Endgeräte (mit verschiedenen Bildschirmgrößen, Grafikbibliotheken und Ressourcen) entwickelt wird (vgl. [LUYTEN 2004, S. 5]). Um den resultierenden Anforderungen und Rahmenbedingungen gerecht zu werden, sind Aspekte der Wiederverwendbarkeit, der Flexibilität und der Plattformunabhängigkeit wesentlich (vgl. [CALVARY U. A. 2003, S. 290]). Dem wiederkehrenden Entwicklungsaufwand für Einzellösungen werden in der Folge *modellbasierte Ansätze*<sup>59</sup> für die Entwicklung von Benutzerschnittstellen entgegengestellt (vgl. [PUERTA 1997, S. 41–42]). Es existieren hierbei *systemzentrierte* und *benutzerzentrierte* Vorgehensweisen und Konzepte, die jeweils eigene charakteristische Merkmale aufweisen und dem Erreichen unterschiedliche Ziele innerhalb des Entwurfsprozesses dienen.

Systemzentrierte Ansätze basieren auf der Idee, eine Anwendung über die Abläufe und Strukturen innerhalb eines Softwaresystems zu beschreiben. Zu diesem Zweck hat die Object Management Group in den 1990er Jahren die Unified Modelling Language (UML) entwickelt (vgl. [FOWLER 2004, KAP. 1]). Die UML ist eine Sprache für die Modellierung von Software, basierend auf den Erkenntnissen und den Erfahrungen aus Softwareprojekten der vorangegangenen Jahrzehnte. Die Sprache definiert in den Bereichen der Struktur-, Verhaltens-, Interaktions- und Implementierungsmodellierung jeweils verschiedene Diagrammart, basierend auf standardisierten Notationen, mit denen die „*Spezifikation, Konstruktion, Visualisierung und Dokumentation von Modellen für Softwaresysteme*“ [OESTERICH 2005, S. 12] erfolgen kann. Mithilfe der Beschreibungsformen können die Struktur und die Funktionsweise der Benutzerschnittstelle auf Basis von (meist) deklarativen Modellen abgebildet werden (vgl. [SCHLUNGBAUM & ELWERT 1996, S. 8–9]). Die Gesamtheit der Modelle, die für eine Interfaceentwicklung eingesetzt werden, beschreibt PUERTA als das *Interface Model* [PUERTA 1997, S. 41].

Benutzerzentrierte Ansätze haben ihren Ursprung hingegen im Bereich der Mensch-Computer-Interaktion. Die Vorgehensweisen werden mit dem Anspruch entwickelt, durch eine

<sup>59</sup> engl. Model-Based User Interface Development, MBUID

modellbasierte Beschreibung, die Wünsche, Ziele und Aufgaben eines Anwenders in der Benutzerschnittstellenentwicklung zu berücksichtigen. Aufgrund der unterschiedlichen Disziplinen sind die Vorgehensweisen durch individuelle Methoden aus den Bereichen der Softwareergonomie (vgl. [ROSSON & CARROLL 2002]), des Software Engineerings (vgl. [CONSTANTINE & LOCKWOOD 1999]) und der Geschäftsprozess-Modellierung (vgl. [HOLT 2009]) gekennzeichnet. Während Softwareentwickler häufig eine technisch geprägte Sicht auf ein benutzerzentriertes Vorgehen besitzen, nehmen Gestalter in ihrer Vorgehensweise überwiegend eine vermittelnde Position zwischen Anwendern und Entwicklern ein. In der Folge dienen für die Analyse und den Entwurf benutzerzentrierter Benutzerschnittstellen individuelle Beschreibungsformen, denen jedoch keine standardisierte Notation zugrunde liegt. Eine Vereinheitlichung der Beschreibungsformen, wie diese von MEMMEL und REITERER für die agile Interfacespezifikation aufgezeigt ist (vgl. [MEMMEL & REITERER 2008, S. 360]), wird durch eine fortwährende Veränderung der Visualisierungssysteme und Interaktionstechnologien erschwert. Daher ist auch in naher Zukunft keine gemeinsame Beschreibungsform absehbar, die aus einer vorhandenen und allgemeingültigen Theorie der Mensch-Maschine-Kommunikation abgeleitet ist. Andererseits werden seit etwa 15 Jahren im Bereich benutzerzentrierter Entwicklungen Entwurfsmuster eingesetzt, um das bestehende Wissen über die Interfaceentwicklung in abstrahierter Form zu transferieren und wiederzuverwenden. Eine allgemeine Definition der Entwurfsmuster im Bereich der Mensch-Computer-Interaktion (vgl. [GRAHAM 2003, S. 1]) führte dazu, dass einerseits eine Vielzahl von Entwurfsmustersammlungen und Mustersprachen existiert, andererseits es nicht eindeutig ist, welche Anforderungen an Muster und deren Notation zu stellen sind. Ungeachtet dessen haben sich Entwurfsmuster im Benutzerschnittstellenentwurf, unabhängig von der jeweils zugrunde liegenden Beschreibungssprache und Notation, etabliert.

Für die system- und die benutzerzentrierten Vorgehensweisen können zusammenfassend zwei wesentliche Aspekte benannt werden, die im Bezug auf die 3D-Interfaceentwicklung zu berücksichtigen sind und in den nachfolgenden zwei Unterabschnitten näher betrachtet werden:

- *Modellbasierte Beschreibung von 3D-Benutzerschnittstellen:* Eine Bereitstellung von Modellen und Methoden, die eine abstrahierte Beschreibbarkeit von 3D-Benutzerschnittstellen bieten, ermöglichen den strukturierten Entwurf, die Übersicht im Gesamtprozess und die Wiederverwendbarkeit von Entwürfen bei der Entwicklung von 3D-Interfaces.
- *Wahrung von Entwurfswissen im 3D-Interfacedesign:* Im Hinblick auf den interaktiven Nutzer-Bild-Dialog sollte gewährleistet sein, dass die zu entwickelten 3D-Benutzerschnittstellen notwendige softwareergonomische Anforderungen erfüllen.

#### 5.1.2.1 Modellbasierte Beschreibung von 3D-Benutzerschnittstellen

Bei einem modellgestützten Entwicklungsansatz dienen deklarative Modelle der formalen Beschreibung eines Interfacedesigns, wobei eine klare Trennung der fachlichen Essenz (die Anwendersicht) von den technischen Details (die Entwicklersicht) bestehen sollte. Ein modellbasiertes Vorgehen muss in der Folge über Beschreibungsmethoden verfügen, die zueinander

in Beziehung stehen, um einen Entwurfsprozess von Benutzerschnittstellen ganzheitlichen zu unterstützen. Ein modellbasierter Ansatz für 3D-Interfaces soll vor allem die folgenden Aspekte im Entwurfs- und Entwicklungsprozess berücksichtigen:

- *Die Trennung von Benutzerschnittstelle und Anwendungslogik:* Im Entwicklungsprozess von Benutzerschnittstellen ist es aus softwaretechnologischer Sicht notwendig, das grafische Interface von der Anwendungslogik zu trennen. Dies ergibt sich einerseits aus dem Umstand, dass im Entwicklungsprozess einer Anwendung, die Umsetzung der Benutzerschnittstelle parallel zur Entwicklung der Anwendungslogik erfolgt oder dieser bereits vorausgegangen ist. Andererseits sind an der Benutzerschnittstellenentwicklung Personen ohne Programmierhintergrund beteiligt, deren Arbeit durch eine Verknüpfung mit der Programmlogik erschwert oder unmöglich wird. Weiterhin sollten die Abhängigkeiten zwischen der Benutzerschnittstelle und der Anwendungslogik minimiert sein, damit Änderungen in der Benutzerschnittstelle keinen Einfluss auf die Anwendungslogik haben.
- *Die Plattformunabhängigkeit:* Die Plattformunabhängigkeit ist ein wichtiges Kriterium bei der Softwareentwicklung, da es die Verfügbarkeit von Anwendungen auf unterschiedlichen Systemen sicherstellt. Im Zusammenhang mit der Entwicklung von 3D-Benutzerschnittstellen wird der Begriff der Plattformunabhängigkeit abstrahiert ausgelegt. Es besteht die Forderung nach einer Beschreibungsform, die von unterschiedlichen Anwendungen interpretiert werden kann. Die Plattformunabhängigkeit intendiert infolgedessen, dass entweder aus einer Beschreibung der Quellcode für Anwendungen auf verschiedenen Plattformen generiert oder durch eine Interpretation von der Zielapplikation eine Benutzerschnittstelle erzeugt werden kann.
- *Deklarative Beschreibung der Benutzerschnittstelle:* Die Beschreibung einer 3D-Benutzerschnittstelle sollte einen deklarativen Charakter aufweisen, da diese Form gegenüber der imperativen Beschreibung den Vorteil hat, dass dieser von unterschiedlichen Personengruppen verstanden wird. Die deklarativen Elemente sollten sowohl die Eigenschaften der Interfaceelemente als auch deren Beziehungen untereinander im Entwurf eines Interface abbilden können.
- *Die Transformation der Beschreibung:* Die Beschreibung einer Benutzerschnittstelle sollte auf anerkannten Standards beruhen und eine Überführung in unterschiedliche Zielformate unterstützen. Durch eine etablierte Beschreibungsgrundlage werden die Voraussetzungen für eine Transformation der Interfacebeschreibung in verschiedene Zielformate verbessert und eine zukünftige vielseitige Nutzung begünstigt.
- *Die Werkzeugunterstützung:* Die Entwicklung von Benutzerschnittstellen ist oftmals aufwendiger als die Entwicklung der Applikationslogik. Um die Entwicklungszeit zu verkürzen, werden Modellierwerkzeuge und GUI-Builder eingesetzt, die in der Regel ein Bestandteil einer Entwicklungsumgebung sind oder begleitend zur Verfügung stehen. Dies ist notwendig, da aufgrund kurzer Entwicklungszyklen es kaum möglich ist, auf die Vorteile einer grafischen Unterstützung zu verzichten. Ein Benutzerschnittstellenentwurf sollte durch Entwicklungs- und Entwurfswerkzeuge erstellt, interpretiert und ebenso eine Beschreibung durch diese generiert werden können.

### 5.1.2.2 Wahrung von Entwurfswissen im 3D-Interfacedesign

Generelle Gestaltungshinweise formulieren Empfehlungen für den Interfaceentwurf, die für einen festgelegten Kontext gültig und nicht auf ein System oder eine Applikation begrenzt sind. Die Anwendung von Gestaltungshinweisen für 2D-Benutzerschnittstellen ist, mit Ausnahme allgemeingültiger Hinweise, aufgrund der Charakteristik von interaktiven 3D-Interfaces nur bedingt gangbar (siehe Unterabschnitt 5.1.1). Für generelle Gesichtspunkte bieten die Gestaltungshinweise, beispielsweise SHNEIDERMANs acht goldene Regeln des Interfacedesigns (vgl. [SHNEIDERMAN 2004, S. 74]), eine Entwurfsunterstützung in einer allgemeingültigen Form. Die Hinweise sind in deren Ausprägungen vielseitig einsetzbar und innerhalb einzelner Phasen des Entwicklungsprozesses, zum Beispiel bei der Konzeption oder der Evaluation, zweckgerichtet anwendbar. Dabei definieren die Hinweise keine formalen Regeln der Gestaltung, weshalb die Entwurfshilfen eine Interpretation benötigen (vgl. [MARIAGE U. A. 2004, S. 503–504]). Eine konkrete Beschreibung bieten indes Entwurfsmuster und Mustersprachen. Diese sind praxisorientiert und in der Folge schnell umsetzbar, da Entwurfsmuster im Unterschied zu Gestaltungshinweisen einen geringen Interpretationsaufwand benötigen. Durch die Darbietung von Lösungen zu wiederkehrenden Problemstellungen bilden Entwurfsmuster den aktuellen Wissensstand in einem spezifischen Bereich ab. Durch Entwurfsmuster aus dem Bereich der Mensch-Computer-Interaktion wird der Entwicklungsprozess von Benutzerschnittstellen, unter ergonomischen Gesichtspunkten, um konkrete Vorlagen und Beispiele bereichert. BORCHERS formuliert diesbezüglich als eine von zwölf Thesen zum Einsatz von Entwurfsmustern im Interaktionsdesign: *“The use of patterns in [software architecture, interaction design and the application domain] can be mapped to most phases of the usability engineering life cycle.”*<sup>60</sup> [BORCHERS 2000A, S. 5]. Ein Muster existiert dabei nicht isoliert, sondern steht in einem Beziehungsnetzwerk mit anderen Mustern; in einer Mustersprache. Der Nutzen einer solchen Sprache ergibt sich hierbei nicht aus der Anzahl der Entwurfsmuster, welche die Sprache bilden, sondern erwächst aus dem kombinierten Einsatz der Muster. Daher weisen einige Mustersprachen modell- beziehungsweise ebenenbasierte Strukturen auf, um die Entwurfsmuster in Abhängigkeit zu den Phasen des Entwurfsprozesses zu formulieren (vgl. [VAN WELIE & VAN DER VEER 2003, S. 528–529]). Die Anwendung der Musterlösungen erfolgt somit entsprechend der im Verlauf der Entwicklung auftretenden Herausforderungen.

### 5.1.3 Motivation für einen modellbasierten und mustergestützten Entwurfsprozess

Vorgehensmodelle und Interfacebeschreibungssprachen für den Interfaceentwurf wurden in den 1980er Jahren für traditionelle Benutzerschnittstellen entwickelt, konnten sich in diesem Bereich jedoch nicht etablieren (vgl. [MYERS U. A. 2000, S. 11]). Der Grund hierfür liegt vor allem in einem Mangel an Einflussmöglichkeiten, über die ein Entwickler bei der faktischen Ausgestaltung des Interface verfügt. Darüber hinaus benötigen die Vorgehensweisen unter Verwendung von Interfacebeschreibungssprachen einen hohen Einarbeitungsaufwand und

<sup>60</sup> „Die Verwendung von Entwurfsmustern in den Bereichen der Softwarearchitektur, des Interaktionsdesigns und der Anwendungsdomäne kann auf die meisten Phasen der Planungs- und Entwicklungsarbeit zur Sicherstellung der Gebrauchstauglichkeit eines Systems abgebildet werden.“



eine eingehende Lernbereitschaft des Entwicklers für deren effektiven Einsatz. Allerdings wird vor allem die Etablierung der Desktop-Metapher der ausschlaggebende Grund sein, der zur Quasi-Standardisierung des WIMP<sup>61</sup>-Interface beitrug. Aufgrund dessen wurden die modellbasierten Ansätze für konventionelle 2D-Benutzeroberflächen nicht weiterentwickelt.

Die Gestaltung und die Entwicklung von 3D-Interfaces unterscheiden sich jedoch von konventionellen 2D-Benutzeroberflächen und motivieren zu einer Wiederaufnahme der modellbasierten Konzepte. Dies liegt zum einen im Fehlen einer Grundmetapher für 3D-Interfaces begründet. Diesbezüglich argumentiert BOWMAN, dass sich eine vergleichbare fundamentale Metapher, aufgrund der unterschiedlichen Interaktionsmöglichkeiten, der vielgestaltigen Ausgabegeräte und der mannigfaltigen Anwendungskontexte, möglicherweise nie etablieren wird (vgl. [BOWMAN u. A. 2004, S. 4–5]). Zum anderen liegt bei 2D-Benutzeroberflächen der Fokus von Layout- und Dialoggestaltung auf der Interaktion des Nutzers mit grafischen Symbolen, während sich die visuellen Elemente und die Interaktionen von 3D-Benutzerschnittstellen durch die geometrische Struktur des Datenraums in Relation zum Anwender und dessen Verhalten manifestieren. Aufgrund dessen ist das Konzept einer ereignisorientierten Benutzerschnittstellenbeschreibung, wie dieses für 2D-Interfaces eingesetzt wird, nur bedingt auf 3D-Benutzerschnittstellen übertragbar. Der Nutzer-Bild-Dialog in interaktiven 3D-Anwendungen muss vielmehr auf einem abstrahierten Niveau spezifiziert werden, der durch eine modellbasierte Beschreibung erreicht wird und in der Folge eine Strukturierung sowie Nachvollziehbarkeit im Entwurfsprozess und eine Konsistenz sowie Wiederverwendbarkeit von 3D-Interfacekonzepten sicherstellt. Eine modellhafte Beschreibung führt allerdings dazu, dass lediglich gleichartige Benutzerschnittstellen erstellt werden können, da deren tatsächliche Ausgestaltung nicht von definierten, faktischen Schnittstellenelementen, wie beispielsweise Schaltflächen, sondern von den Elementen des Objektraums und der Veränderbarkeit des Bildraums festgelegt ist.

Die Transformation der Modellbeschreibung einer 3D Benutzerschnittstelle in die tatsächliche Ausgestaltung einer interaktiven 3D Anwendung kann im Zuge des Entwurfsprozesses auf Basis einer Programmiersprache erfolgen. Dies kann jedoch den strukturierten und ineinandergreifenden Übergang vom Entwurf in die Realisierung behindern, da die Verständlichkeit von Quellcode für Personengruppen ohne Programmierkenntnisse eingeschränkt ist. Während für die Realisierung einer Benutzerschnittstelle auf Basis einer Programmiersprache infolgedessen der Designer den Interfaceentwurf zunächst weitestgehend unabhängig vom Entwickler realisiert, ermöglichen Interfacebeschreibungssprachen die gemeinsame Festlegung der Struktur und des Aufbaus einer Benutzerschnittstelle und darüber hinaus die direkte Anbindung an die Programmlogik durch den Entwickler. Mithilfe des Parsens der Interfacebeschreibung erfolgt die automatische Erzeugung von Quellcode oder einer Interfaceinstanz aus der Beschreibung, wodurch eine von allen Personengruppen nachvollziehbare Beschreibungsform für die Ausgestaltung der 3D-Benutzerschnittstelle in einer interaktiven Anwendung dient. Eine daraus resultierende Bestrebung nach einer Neuentwicklung einer proprietären Beschreibungssprache für 3D-Benutzerschnittstellen ist jedoch nicht zielführend, da verschiedene Beschreibungssprachen existieren, die den Entwurf eines

<sup>61</sup> Englischsprachiges Akronym, das für „Windows“, „Icons“, „Menus“, „Pointing Device“ steht

Datenbildes in einer 3D-Benutzerschnittstelle ermöglichen und durch die Beschreibung des Dialogbildes ergänzt werden können.

Ein modellbasierter Interfaceentwurf in Abhängigkeit vom Datenbild ist mit der bestehenden Vielgestaltigkeit an Darstellungstechniken und der interaktionsförderlichen Veränderlichkeit des Bildraums mannigfaltig und bedarf eines weitgehend manuellen Gestaltungsprozesses, um die Umsetzung von Visualisierungszielen im Nutzer-Bild-Dialog sicherzustellen. Da jedoch auch ein modellgestützter Entwurfsprozess, auf Basis von Modellierungs- und Auszeichnungssprachen, die Qualität eines Interface a priori nicht sicherstellt, ist es wesentlich, das modellbasierte Vorgehen mit benutzerzentrierten Ansätzen zu erweitern beziehungsweise zu ergänzen (vgl. [PETRASCH 2007, S. 7–8]). Entwurfsmuster bilden hierzu ein mächtiges Konzept für die Erfassung von Entwurfswissen über effizient gestaltete Benutzerschnittstellen. Zur Unterstützung des Entwurfsprozesses sollten Mustervorlagen eingesetzt werden, die einen funktionalen und darüber hinaus praktikablen Benutzerschnittstellenaufbau beschreiben (vgl. [BEALE & BORDBAR 2005, S. 2]). Dadurch werden nutzerzentrierte Gestaltungshinweise kommuniziert und gleichzeitig aus einer technischen Sichtweise heraus aufgezeigt (vgl. [MAHEMOFF & JOHNSTON 1998, S. 28]). Dies kann erreicht werden, indem die zur Erfüllung einer Nutzeraufgabe einzusetzenden Interfaceelemente und deren Anordnungen sowie deren Abhängigkeiten zueinander in der Mustervorlage beschrieben werden. MAHEMOFF und JOHNSTON stellen heraus, dass diese Form von Entwurfsmustern vor allem von Gestaltern und Entwicklern der Benutzerschnittstellen eingesetzt werden können (vgl. [MAHEMOFF & JOHNSTON 1998, S. 29]). Die Autoren befürworten den Einsatz dieser Entwurfsmusterart in der Interfaceentwicklung wie folgt: *“its proximity to existing techniques suggests that it would be a step in the right direction for developers who might otherwise disregard the human factor”*<sup>62</sup> [MAHEMOFF & JOHNSTON 1998, S. 29].

## 5.2 Metamodelle für den 3D-Interfaceentwurf

In den Gestaltungshinweisen für 3D-Interfaces (siehe Abschnitt 3.4) wird eine Interdisziplinarität im Spezifikations- und Entwurfsprozess von 3D-Benutzerschnittstellen insistiert. Dies begründet sich aus dem Umstand, dass die Akteure im Entwurfsprozess über individuelle Sichtweisen und Ausdrucksmittel verfügen, um das Ziel des Entwurfs und der Beschreibung von ergonomischen und gebrauchstauglichen 3D-Benutzerschnittstellen zu erreichen. Bei einer Zusammenarbeit von Akteuren mit unterschiedlichen Kompetenzen und verschiedenen Qualifikationen werden Komplikationen deutlich, die sich im Mangel an kollaborativen Methoden und Beschreibungsformen vergegenwärtigen. In der Folge dominieren textbasierte Spezifikationsdokumente als Kompromisslösung den interdisziplinären Entwurfsprozess (vgl. [MEMMEL & REITERER 2008, S. 357]). Da jedoch textuelle Beschreibungen häufig einen erheblichen Interpretationsspielraum zulassen, werden der Entwurf von 3D-Benutzerschnittstellen und die Entwicklung von interaktiven Anwendungen durch eine ungenaue Spezifikation erschwert. In der Folge ist eine wechselseitige Überführung von abstrakten Konzepten und detaillierten Entwürfen sowie Interfacebeschreibungen nicht gegeben. Ein strukturierter und

<sup>62</sup> „Die Nähe zu bestehenden Techniken lässt vermuten, dass diese ein Schritt in die richtige Richtung für die Entwickler sind, die ansonsten den menschlichen Faktor unberücksichtigt lassen würden“

wechselseitig überführbarer Interfaceentwurf ist indes erforderlich, um den Entwicklungsprozess effizient zu vollziehen (vgl. [METZKER & REITERER 2002, S. 27–28]). Hierbei dienen Vorgehensmodelle einerseits dazu die Erfordernisse in Anforderungen zu überführen, andererseits eine Spezifikation in ein Interfacedesign zu übersetzen, wodurch der einfache Wechsel von abstrakter Beschreibung (Text), modellhafter Skizzierung (Modellierung) und Interfacekonstruktion (Beschreibung) erreicht wird.

### 5.2.1 UML als Modellierungssprache

Die *Unified Modelling Language* (UML) ist eine grafische Modellierungssprache zur Spezifikation, Konstruktion und Dokumentation von Softwaresystemen und Anwendungen. Die Sprache wird von der Object Management Group (OMG) entwickelt und ist durch die ISO<sup>63</sup> standardisiert (vgl. [OMG]). Die UML bietet verschiedene grafische Notationstechniken für die Erstellung von Modellen für statische Strukturen und dynamische Abläufe zur Unterstützung der objektorientierten Softwareentwicklung. Infolge dieser Ausrichtung hat sich die Sprache in den Bereichen der softwaretechnischen Analyse und des Entwurfes etabliert und wird für fast alle Anwendungsdomänen und auf annähernd jeder Softwareplattform eingesetzt. Die Eigenschaften und Notationen der UML werden an dieser Stelle nicht weiterführend beschrieben, da die verschiedenen Modelltypen und unterschiedlichen Notationen in der Literatur (siehe [FORBRIG 2002; FOWLER 2004]) umfassend erläutert werden.

Die Bestrebung der Object Management Group, die UML für heterogene Anwendungsdomänen zur Verfügung zu stellen, beschränkt die Sprache gleichzeitig in dessen Eigenschaft spezielle Bereiche und Aspekte detailliert zu modellieren, da hierbei die Beschreibung in einer domänenspezifischen Form notwendig ist. Aufgrund der bestehenden Syntax und Semantik der UML können beispielsweise kennzeichnende Eigenschaften eines interaktiven Systems nicht vollständig beschrieben werden, da die UML-Elemente zu allgemeingültig definiert sind und ein konkretes Merkmal eines Systems in der Folge nicht eindeutig charakterisiert werden kann. Um diesen Beschränkungen zu begegnen, bietet die UML-Spezifikation ein Metamodell zur Profildefinition, durch das die Sprache an fachliche oder technische Domänen angepasst werden kann. Durch die *UML-Profile* können domänenspezifische Sprachen definiert werden, indem eine Spezialisierung ausgewählter UML-Sprachelemente erfolgt und neue Definitionen sowie Restriktionen auf diesen eingeführt werden.

UML-Profile basieren somit auf dem UML-Metamodell und bilden in der Folge Elemente beziehungsweise konkrete und spezifische Sprachen der UML-Familie. Die folgenden Eigenschaften von UML-Profilen (vgl. [OMG]) motivieren deren Einsatz für den 3D-Benutzerschnittstellenentwurf:

- Die Festlegung einer Terminologie, die an den domänenspezifischen Kontext des 3D-Interfaceentwurfes, unter Einbindung linearer und nichtlinearer Darstellungstechniken, angepasst ist.
- Das Hinzufügen von Bedeutungseinheiten, welche durch die bestehende Semantik der UML nicht abgebildet werden kann.

<sup>63</sup> International Organization for Standardization

- Die Modelle von 3D-Benutzerschnittstellen können in bestehenden UML-Werkzeugen erstellt und modifiziert werden.
- Die UML-Notationen bieten eine etablierte und standardisierte Basis für die Modellierung, weshalb keine neue Sprache durch den Entwickler erlernt werden muss.

### 5.2.2 PLML als Musterdefinitionssprache

In [MAHEMOFF & JOHNSTON 1998] werden verschiedene Ausprägungen von Entwurfsmustern und Mustersprachen identifiziert, analysiert und diese in Bezug auf das Interaktionsdesign von Benutzerschnittstellen in aufgaben-, nutzer-, interfacegestaltende- und systembezogene Entwurfsmuster unterschieden. Durch diese Aufgliederung werden die Vielgestaltigkeit und die verschiedenen Unterstützungsmöglichkeiten der Entwurfsmuster deutlich. Für die Beschreibung von Entwurfsmustern wird daher eine dem Kontext des 3D-Interfaceentwurfes angepasste Granulierung und Attribuierung gewählt. Während die Entwurfsmuster ALEXANDERS für die Architektur (vgl. [ALEXANDER U. A. 1977]) nur den Musternamen, das Problem, den Kontext, die Idee einer Lösung und die Abhängigkeiten umfassten, sind im Bereich interaktiver 3D-Interfaces weitere Merkmale als Beschreibungskriterien zweckmäßig. Aufbauend auf den Kriterienkatalogen aus der Mensch-Computer-Interaktion (vgl. [DUYNE U. A. 2003, S. 20 FF.]) und der Softwareergonomie (vgl. [FOLMER U. A. 2003, S. 72]) werden die Attribute für die Entwurfsmuster der 3D-Interfaceentwicklung nachfolgend festgelegt. Dabei wird die *Pattern Language Markup Language (PLML)* zugrunde gelegt (vgl. [FINCHER & FINLAY 2003]), die zur Attribuierung von Entwurfsmustern eine Elementmenge aus dem Bereich der Mensch-Maschine-Kommunikation bereitstellt (vgl. [PLML]):

- *Der Name:* Jedes Muster besitzt einen prägnanten Namen, der die Problemstellung aufzeigt.
- *Das Problem:* Die zu lösende Aufgabe, zu dessen Bewältigung das Muster eingesetzt werden soll, wird aufgeführt.
- *Der Problemkontext:* Ein Kontext wird beschreiben, wenn das Problem unter verschiedenenartigen Vorbedingungen auftreten kann.
- *Die Kategorie:* Da die Lösungsvorlagen in unterschiedlichen Phasen des Entwurfes angewendet werden, wird die Zuordnung des Musters zum entsprechenden Verfahrensschritt benannt.
- *Der Lösungsvorschlag:* Eine Beschreibung zur Lösung des Problems wird in Form einer abstrakten Vorlage gegeben.
- *Das Diagramm:* Die schematische Erläuterung dient der Beschreibung von Details der Mustervorlage in einer formalen Darstellung unter Zuhilfenahme der UML.
- *Die Vor- und Nachteile:* Wenn ein Problem in einer bestimmten Weise gelöst wird, kann sich das vorteilhaft oder nachteilig auf andere Aspekte des zu entwickelnden Gesamtsystems auswirken. Daher werden sowohl Vor- als auch Nachteile aufgeführt.

- *Der Verwendungskontext:* Die Angabe einer in diesem Sinne guten Umsetzung zeigt die Anwendbarkeit des jeweiligen Musters in einem konkreten Kontext oder in einer praktischen Anwendung.
- *Die Verbindungen:* Die Verknüpfungen zu anderen Entwurfsmustern werden angegeben, um Zusammenhänge und Abhängigkeiten aufzuzeigen.
- *Die Illustration:* Zur Verdeutlichung der Anwendung eines Entwurfsmusters wird optional eine Benutzerschnittstelleninstanz in Form einer schematischen Darstellung oder eines Benutzerschnittstellenschnappschusses dargeboten.

### 5.2.3 XML als Beschreibungssprache

Eine Vielzahl entwickelter Beschreibungssprachen für unterschiedliche Aspekte des Benutzerschnittstellenentwurfes basiert auf der Auszeichnungssprache XML (Extensible Markup Language). Das World Wide Web Consortium (W3C) entwickelte XML als eine Untermenge der *Standard Generalized Markup Language (SGML)* für die Darlegung hierarchisch strukturierter Daten und der Charakterisierung von Datenelementen (vgl. [XML]). Die durch das W3-Konsortium herausgegebene Spezifikation definiert XML als Metasprache, auf deren Basis durch strukturelle und inhaltliche Einschränkungen Sprachen definiert werden können. Diese Einschränkungen werden durch Schemasprachen ausgedrückt. In einer solchen Sprache werden die Elementnamen und die Attribute definiert, wodurch die Deklaration des Sprachumfangs einer konkreten Sprache erfolgt. Somit wird eine einheitliche Form und Struktur von XML-Dokumenten sichergestellt. Die Vorteile von XML als Meta-Auszeichnungssprache sind in vielen Anwendungsdomänen und Projektkontexten belegt. Beispielsweise ist X3D eine XML-basierte Beschreibungssprache für 3D-Modelle (vgl. [BRUTZMAN & DALY 2007]), *InTML* eine Erweiterung der X3D-Beschreibungssprache für Interaktionstechniken (vgl. [FIGUEROA U. A. 2002]) und der *CONTIGRA-Ansatz* eine Beschreibungsform zur deklarativen und interdisziplinären Erstellung von 3D-Anwendungen (vgl. [DACHSELT 2001]). Einige Vorteile der Extensible Markup Language (vgl. [XML]) werden nachfolgend benannt, um den Einsatz für die Beschreibung von 3D-Benutzerschnittstellen zu motivieren:

- Die Beschreibungssprache bietet eine Plattformunabhängigkeit.
- Die Elemente in einem XML-Dokument können hierarchisch strukturiert werden, wodurch gerichtete azyklische Graphen – einem Szenengraphen entsprechend – umgesetzt werden können.
- XML ist eine Sprache in einem menschenlesbaren Format und bietet die Möglichkeit einer automatisierten Verarbeitung.
- XML ist ein etablierter Standard der von Quelltexteditoren und Entwicklungsumgebungen unterstützt wird, wodurch ein werkzeuggestützter Entwurfsprozess möglich ist.

Der Sprachumfang zur Beschreibung einer 3D-Benutzerschnittstelle soll aus einer erweiterbaren Menge an Elementen bestehen. Dafür sind eine formale Strukturbeschreibung der Elemente und deren Zusammenhänge innerhalb eines XML-Dokumentes notwendig. Hierfür

existiert einerseits die *Document Type Definition (DTD)*, andererseits mit *XML Schema* eine Weiterentwicklung von DTDs für die Deklaration von Struktur und Inhalt. XML Schema besitzt gegenüber DTDs Vorteile, die nachfolgend aufgeführt sind und den Einsatz von XML Schema für die Festlegung der Art und Struktur der XML-Dokumente begründet:

- XML Schema verfügt über eine Vielzahl vordefinierter Datentypen und bietet gleichzeitig die Möglichkeit eigene Datentypen zu definieren.
- Die Sprache verwendet die Syntax von XML, wodurch das Erlernen einer neuen Syntax entfällt.
- Elemente können durch Reihung, Konjunktion oder Auswahl flexibel strukturiert werden.
- Bestehende komplexe Elemente können um Elemente oder Attribute erweitert beziehungsweise eingeschränkt werden.

### 5.3 Das Mosalk-Vorgehen für den Interfaceentwurf

Die Verwirklichung einer 3D-Benutzerschnittstelle besitzt zunächst einen Planungs-, daran anschließend einen Entwurfs- und abschließend einen Realisierungscharakter. Es handelt sich um ineinander übergehende Phasen, deren Resultat eine 3D-Benutzerschnittstelle, einschließlich der Anweisungen für den maschinellen Ablauf des grafischen Interface in der interaktiven 3D-Anwendung, ist. In diesem Entwurfsprozess ist es notwendig, zum einen möglichst exakte Beschreibungen der gegebenen Anforderungen sicherzustellen, zum anderen die konzeptionellen Entwürfe und detaillierten Festlegungen eines 3D-Interface eindeutig zu formulieren. Für eine informationelle Bestimmung der Anforderungen an eine 3D-Benutzerschnittstelle muss einerseits der Informationsgehalt über den diese verfügen soll, andererseits die Formulierungs- und Beschreibungserfordernisse für den Entwurfsprozess identifiziert werden. Der inhaltliche Fokus und der Nutzungskontext eines 3D-Interface sowie die Bedingungen, Eigenschaften und Parameter legen dessen informationsstrukturellen Rahmen fest. Anhand der am Entwicklungsprozess beteiligten Personengruppen und deren Sicht auf eine Benutzerschnittstelle lassen sich Folgerungen in Bezug auf den Umfang und die Struktur der Spezifikationsinhalte (Informationsstrukturen) und die einzusetzenden Beschreibungsformen (Sprachkonstrukte) ziehen und ferner folgende Leitfragen ableiten (vgl. [CHLEBEK 2002, S. 3]):

- Welche Inhalte hat die Anwendung darzubieten und welche Abläufe aus fachlicher Sicht zu erfüllen?
- Wie sind Inhalte und Abläufe in der Anwendung strukturell verknüpft?
- Wie wird eine konsistente visuelle Darstellung der Inhalte erreicht?
- Welche Sichten auf den Anwendungsinhalt sind zu verwirklichen?
- Wie wird der Anwender in seinem Arbeitsablauf und bei einzelnen Handlungen unterstützt?
- Nach welchen Regeln erfolgen die Darstellungsabläufe?

Aus diesen Leitfragen lässt sich eine fachliche Gliederung der Interfacespezifikation in unterschiedlichen Betrachtungsweisen ableiten:

- *Die Anforderungssicht* – die Anforderungen bilden den Ausgangspunkt für den Aufbau und die Elemente der Benutzerschnittstelle sowie die semantischen Zusammenhänge in einem gegebenen Kontext.
- *Die Struktursicht* – die Festlegung des strukturellen Aufbaus einer Benutzerschnittstelle basiert auf den Anforderungen an die interaktive Anwendung. Die Ablaufstruktur bestimmt den bildgestützten Kommunikationsprozess.
- *Die Schnittstellensicht* – der Aufbau der grafischen Benutzerschnittstelle legt die visuelle Ausprägung des interaktiven Systems fest, deren Umsetzung den Anforderungen folgt und in Abhängigkeit vom strukturellen Aufbau festgelegt ist.
- *Die Darstellungssicht* – der bildstrukturelle Aufbau unter Einbeziehung eines konkreten Objektraums und die infolgedessen konkrete Ausgestaltung des Bildraums umfasst die Darstellungssicht auf eine Benutzerschnittstelle.
- *Die Verhaltenssicht* – die Festlegung des Verhaltens behandelt die Ereignisse und die Reaktionen des Systems auf die Aktionen des Anwenders unter Einbeziehung der Elemente des Objekt- und des Bildraums.

Die Abbildung 25 zeigt eine schematische Darstellung, welche die aufgezeigten Sichten im Hinblick auf eine 3D-Interfacespezifikation strukturiert und die Abhängigkeiten zwischen Ablaufstruktur, grafischer Ausgestaltung und den zugrunde liegenden Anforderungen schematisch verdeutlicht. Die informationsstrukturellen Sichten dienen als Grundlage für die Entwicklung einer Verfahrensweise, die das Beziehungsgeflecht zwischen Visualisierungen, bildgestützter Interaktion, Datenbildelementen, bildstrukturellen Eigenschaften und zugrunde liegenden Anforderungen in einen methodischen Entwurfsablauf binden soll. Dazu muss zunächst ein Vorgehensmodell mit qualifizierten Methoden definiert werden, da für den Entwurfsprozess von 3D-Benutzerschnittstellen kaum konkrete Vorgaben über die Art und die Anzahl der Modelle und Beschreibungen sowie deren Verknüpfungen existieren. Gegenüber einer direkten Umsetzung eines Designentwurfes in eine implementierte Benutzerschnittstelle bietet ein modellgestützter Interfaceentwurf in der Regel eine Vereinfachung der Entwicklung, da auf diese Weise die bestehenden Herausforderungen und Rahmenbedingungen sowie die potenziellen Lösungen im Entwurfsprozess aufgenommen, gesammelt und organisiert werden können (vgl. [PUERTA 1997, S. 40]). Gleichzeitig fungiert eine modellbasierte Interfaceentwicklung als ein „Prüfprotokoll“, da hierbei die Anforderungen festgehalten, die Schritte im Entwurfsprozess dokumentiert und die entwickelten Lösungen dargelegt sind. In der Folge ist die Rückverfolgung ausgehend von den Resultaten zu den Anforderungen möglich, wodurch das Ergebnis eines Entwurfsprozesses reflektiert werden kann. Eine wesentliche Voraussetzung für den erfolgreichen Einsatz von Modellen im 3D-Interfaceentwurf ist dabei die Passung von Vorgehensweise und Entwurfsaufgabe.

Der nunmehr vorgestellte Ansatz der »modellbasierten 3D-Interfacekomposition« (Mosalk) beruht auf dem Bestreben, eine Benutzerschnittstelle in Form von plattformunabhängigen,



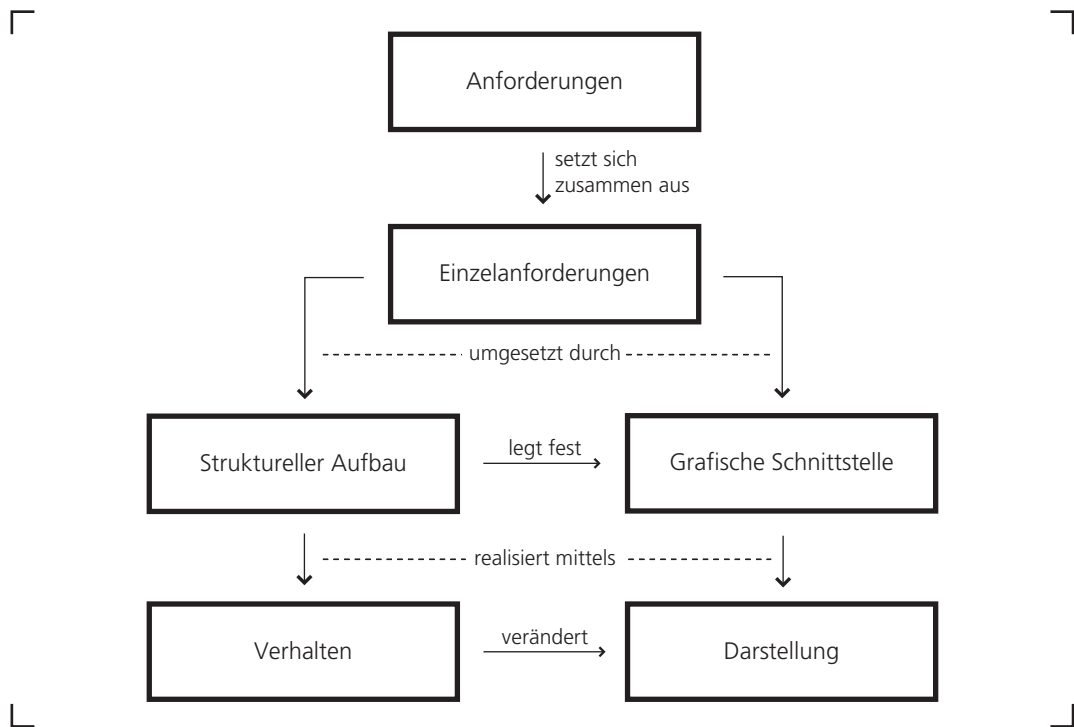


Abbildung 25: Sichtweisen auf die Interfacespezifikation von 3D-Anwendungen

wiederverwendbaren Modellen zu formulieren, um aus dieser Beschreibung wiederum den plattformspezifischen Quelltext zu erstellen. Aufgrund einer bisher fehlenden Referenzarchitektur bei der modellbasierten Entwicklung von Benutzerschnittstellen und dem mangelnden Konsens über die Verwendung bestehender Vorgehensmodelle (vgl. [SHAER U. A. 2008, S. 3951–3952; LUYTEN U. A. 2004]), erfolgte im Rahmen des europäischen CAMELEON Projektes [CAMELEON SIG] die Konzeption einer Meta-Architektur für den Interfaceentwurf. Ein wesentliches Ergebnis dieses Projektes ist das *CAMELEON Reference Framework* (vgl. [CALVARY U. A. 2003, S. 293 ff.]), das sich in den letzten Jahren im Bereich der modellbasierten Benutzerschnittstellenentwicklung etabliert hat. Da es sich hierbei primär um eine Referenzarchitektur zur Klassifizierung von Ansätzen für die modellbasierte Interfaceentwicklung handelt, gibt es keine Softwareimplementierung des Vorgehens oder spezielle Softwarewerkzeuge zur Unterstützung für dessen tatsächliche Anwendung. Das CAMELEON Reference Framework basiert in einer vereinfachten Form auf vier Schichten, die mit *Aufgaben und Konzepte*, *Abstrakte Benutzerschnittstelle*, *Konkrete Benutzerschnittstelle* und *Finale Benutzerschnittstelle* bezeichnet sind (vgl. [VANDERDONCKT U. A. 2004, S. 3]) und für die Mosalk-Vorgehensweise als konzeptionelle Basis dienen. Das Mosalk-Vorgehen für den Entwurf von 3D-Benutzerschnittstellen wird an die vier Abstraktionsebenen angelehnt und wird in Abbildung 26 hinsichtlich des strukturellen Aufbaus schematisch dargestellt. Das Vorgehen unterteilt sich in die drei Phasen *Spezifikation*, *Modellierung* und *Implementierung*, wobei innerhalb dieser Phasen jeweils Interfacebeschreibungen mit unterschiedlichem Konkretisierungsgrad entwickelt werden. Die Ebenen innerhalb der Phasen sind Bezug nehmend auf die Referenzarchitektur mit *konzeptionell*, *abstrakt*, *konkret* und *final* bezeichnet.

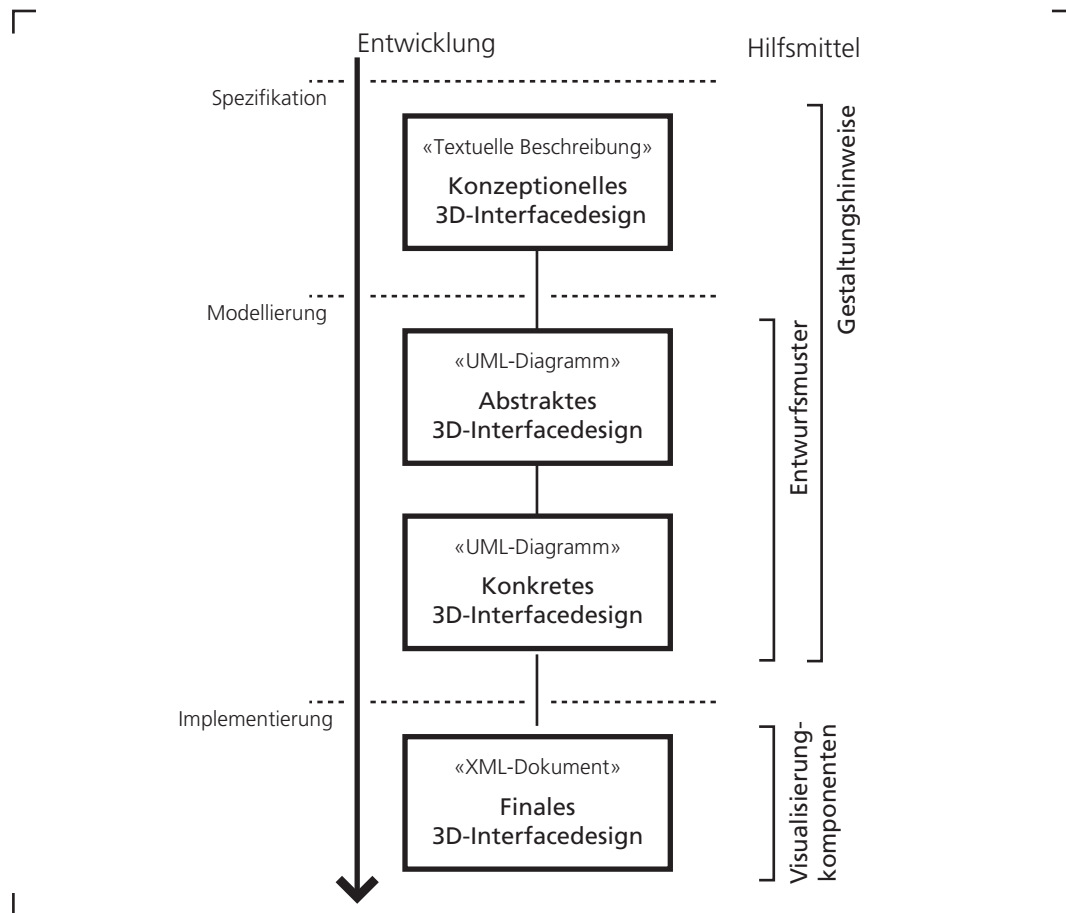


Abbildung 26: Das Mosalk-Vorgehen für den Entwurfsprozess von 3D-Interfaces sowie die vorgehensbegleitenden Hilfsmittel

Die initiale Phase des Vorgehens bildet die Spezifikation, in der die Erstellung des konzeptionellen Interfacedesigns durch eine Identifikation von Nutzungsanforderungen erfolgt. Hierbei handelt es sich um Anforderungen an die effiziente Nutzung eines Systems, die primär aus dem Nutzungskontext, in dem die interaktive Anwendung eingesetzt wird, hergeleitet werden müssen. Mithilfe der identifizierten und formulierten Anforderungen wird in der Modellierungsphase zunächst ein abstraktes Interfacedesign auf Basis eines UML-Profiles erstellt. Dadurch wird die Anforderungsspezifikation in ein abstraktes 3D-Interfacemodell überführt. Das abstrakte Modell wird anschließend durch ein konkretes 3D-Benutzerschnittstellenmodell präzisiert. Für diesen Entwurfsschritt wird ein erweitertes UML-Profil bereitgestellt. Die Modellierung des abstrakten und des konkreten Interfacedesigns wird durch die Bereitstellung von Entwurfsmustern unterstützt. Dabei werden einerseits Aspekte der Gebrauchstauglichkeit adressiert, andererseits Gestaltungshinweise, die in Abschnitt 3.4 formuliert sind, in den Entwurfsprozess eingebunden. Das konkrete 3D-Interfacedesign wird in der Implementierungsphase in das finale 3D-Interfacedesign überführt. Das Resultat dieser abschließenden Phase beschreibt eine Benutzerschnittstelle als Quelltext. Dabei dient eine XML-basierte Beschreibungssprache dazu, das konkrete Interfacemodell in einer maschinenlesbaren Form zu spezifizieren. Resultierend aus dem Parsen des XML-Dokumentes kann die finale Benutzerschnittstelle in einer interaktiven 3D-Laufzeitumgebung verbildlicht werden.

## 5.4 Die Spezifikationsphase

Für die Entwicklung eines 3D-Interface sind in der initialen Phase des Entwurfsprozesses zwei Aspekte wesentlich. Einerseits ist festzulegen, in welchem Kontext eine Benutzerschnittstelle eingesetzt werden soll, andererseits welche Funktion diese zu erfüllen hat. Die Identifikation der Anforderungen an eine 3D-Benutzerschnittstelle erfolgt dabei durch ein Explizieren von Nutzungsanforderungen. Nach MEMMEL U. A. sind diese Anforderungen nicht äquivalent zu fachlichen Anforderungen, welche vor allem Vorgaben darstellen, die ein fachlich korrekt erzieltes Arbeitsergebnis mithilfe des interaktiven Systems sicherstellen sollen. Nutzungsanforderungen sind auch keine Nutzeranforderungen, sondern Forderungen in Bezug auf den effizienten Gebrauch eines Systems, die primär aus dem Nutzungskontext, in dem das interaktive System eingesetzt wird, hergeleitet werden müssen sowie die Grundsätze und Regeln der Dialoggestaltung berücksichtigen (vgl. [MEMMEL U. A. 2008, S. 45]). Daher ist zu Beginn des Entwurfsprozesses zunächst eine Anforderungsspezifikation notwendig, wodurch die Erfordernisse, die für ein System bestehen, in Einzelanforderungen segmentiert werden.

Ein 3D-Interface, welche die gestellten Nutzungsanforderungen erfüllt, ist kontextabhängig, oftmals komplex und durch eine Vielzahl von domain-, daten- und nutzerspezifischen Vorgaben und Festlegungen geprägt. Im Entwurfsprozess ist es essenziell, die gegebenen Erfordernisse zu identifizieren und für die Beschreibung in ein konzeptionelles Interfacedesign zu übersetzen. Für eine solche Überführung müssen gegebene Erfordernisse aus dem „Problemraum“ in einen „Lösungsraum“ transferiert werden. Dieser Vorgang ist in der Regel durch einen häufigen Wechsel zwischen abstrakten Beschreibungen und Designvisionen geprägt. Die am Entwurfsprozess beteiligten Disziplinen bedienen sich für eine strukturierte Aufarbeitung der Nutzungsanforderungen individueller Methoden und Vorgehen. Während Softwareentwickler dabei eine technische Sicht auf die Anforderungen besitzen, nehmen Interaktionsdesigner einen gestalterischen Standpunkt ein, wobei dies vorwiegend auf Basis einer Interpretation der Aussagen von Domänenexperten und Entwicklern erfolgt. Die Qualität der Benutzerschnittstelle sicherzustellen ist dabei das übergreifende Ziel. Dies führt dazu, dass im Bereich des *Usability Engineering* beispielsweise Personas und Nutzerszenarien für die Anforderungsermittlung eingesetzt werden (vgl. [ROSSON & CARROLL 2002]), während für das Software Engineering vorwiegend Anwendungsfälle und Nutzerrollen identifiziert werden (vgl. [CONSTANTINE & LOCKWOOD 1999]). Für die Geschäftsprozessmodellierung kommen wiederum Geschäftsanwendungsfälle und Geschäftsvisionen für die Identifikation zum Einsatz (vgl. [HOLT 2009]). Im Entwurfsprozess grafischer Interfaces können durch die unterschiedlichen Ausprägungen Probleme auftreten, wenn die beteiligten Akteure ihre Anforderungen nicht ausreichend zum Ausdruck bringen. Dies ergibt sich unter anderem daraus, dass eine Reihe von Arbeitsprozessen innerhalb des Benutzerschnittstellenentwurfes unbewusst ablaufen, weshalb deren Dokumentation häufig in der Anforderungserfassung fehlt. Anforderungsspezifikationen sind in der Folge nicht interdisziplinär nachvollziehbar und Unentschiedenheiten nicht offenkundig. Für die Durchführung des Spezifikationsprozesses ist daher die Professionalität eines menschlichen „Editors“ wesentlich, da dieser die bildgestützte Interaktion zwischen den potenziellen Nutzern und dem System hinsichtlich der zu erfüllenden Nutzungsanforderungen zu spezifizieren hat (vgl. [MEMMEL U. A. 2008, S. 47]).

Die Herausforderung besteht vor allem im methodischen Erkennen und Darlegen von impliziten Erfordernissen im Nutzungskontext. In [MEMMEL 2009, S. 52 ff.] werden in der Folge verschiedene Vorgehensweisen und Werkzeuge für eine Anforderungsermittlung analysiert und die Methoden identifiziert, welche fachübergreifend eingesetzt werden können, um die Festlegung von Nutzungsanforderungen für alle beteiligten Personengruppen nachvollziehbar und transparent zu gestalten (vgl. [MEMMEL 2009, S. 136 ff.]). Darauf aufbauend erfolgen für die Erzeugung eines konzeptionellen 3D-Interfacedesigns im Mosalk-Vorgehen zunächst die Identifikation und die Beschreibung von Anwendungsszenarien, die ein interdisziplinär eingesetztes Hilfsmittel darstellen (vgl. [BARBOSA & PAULA 2003]). Die Szenarien dienen der Erfassung von Rahmenbedingungen, Handlungsabläufen und Darstellungsweisen sowie der Beschreibung von potenziellen Anwendern und deren verbundenen Visionen von der bildgestützten Interaktion mit der Benutzerschnittstelle. Die Szenarien werden dazu in Einzelanforderungen überführt, welche die Benutzerschnittstelle zur Verwirklichung des Szenarios zu erfüllen hat. In diesem ersten Entwurfsschritt bleibt das zugrunde liegende System weitestgehend abstrakt, um Designer und Entwickler gleichermaßen nicht durch technologische Restriktionen zu beschränken. Die Beschreibungen erfolgen in natürlicher Sprache, da eine andere Repräsentation als diese eine Kommunikation von Ideen und Konzepten sowie den Gedankenaustausch einschränken beziehungsweise erschweren würde. Diese zunächst informelle Beschreibung des Interface dient folglich der Verbesserung der Kommunikation und der Vereinfachung der Zusammenarbeit. Gleichzeitig kann und soll an dieser Stelle jedoch der situationsbedingte Einsatz von formalen Notationen, Regeln und Methoden, wie diese in [DIX u. A. 2003, KAP. 17] aufgezeigt sind, nicht explizit ausgeschlossen sein. Die aus mehreren – auch verbundenen – Szenarienbeschreibungen abgeleiteten Einzelanforderungen bilden die 3D-Interfacespezifikation. Durch diese Kapselung ist die Nachvollziehbarkeit trotz einer textuellen Beschreibung in einer strukturierten Form möglich. Gleichzeitig ist deren Erweiterbarkeit durch das Hinzufügen weiterer Einzelanforderungen gewährleistet. Die Menge der segmentierten Nutzungsanforderungen stellen nunmehr die Grundlage für Modellierung der 3D-Benutzerschnittstelle dar.

## 5.5 Die Modellierungsphase

Im Verlauf des Entwurfsprozesses folgt auf die Anforderungsspezifikation die Modellierungsphase. Dabei steht der Entwurf der bildgestützten Interaktion zwischen Anwender und System mit dem Ziel im Vordergrund, eine gebrauchstaugliche Benutzerschnittstelle für den gegebenen Kontext zu entwickeln. Mit einem Fokus auf die Beschreibung von 3D-Interfaces in einer modellhaften Form, werden dabei die strukturellen Eigenschaften der Bildräume und der Aufbau des Dialogbildes, respektive der bildgestützten Interaktion, fokussiert. Weitergehende Aspekte der Mensch-Computer-Interaktion, beispielsweise die Festlegung von Techniken für die lexikalische oder die syntaktische Interaktion (vgl. [DIX u. A. 2003, S. 545]), sind indes in der Modellierungsphase des Mosalk-Vorgehens zurückgestellt. Aus dieser Eingrenzung resultiert eine Reduzierung der inkludierten Interfacebestandteile, wodurch angepasste Modelle für den bildgestützten 3D-Interfaceentwurf entwickelt werden können.

Die Modellierungsphase wird in zwei Abstraktionsebenen unterschieden; das abstrakte und das konkrete 3D-Interfacedesign. Da in der Spezifikationsphase die Nutzungsanforderungen als Einzelanforderungen definiert werden, findet auf der abstrakten Modellierungsebene die Verknüpfung von Anforderungsspezifikation und Interfaceentwurf statt. Hierzu wird der interaktive Kommunikationsprozess zwischen Anwender und System skizziert, wobei für die Beschreibung der Dialogbilder die Bildstrukturen und die Regeln für deren Veränderlichkeit im 3D-Interface definiert werden. Im konkreten Interfacedesign wird hingegen die Vergegenständlichung des abstrakten Entwurfes in Bezug auf die einzelnen Bildräume und deren Verknüpfungen fixiert. Durch diesen Modellierungsschritt steigert sich der Detailgrad der Interfacebeschreibung, während die Kontextunabhängigkeit geringer wird. Bei der Festlegung geeigneter Beschreibungsnotationen innerhalb der Modellierungsphase werden zwei UML-Profil eingeführt, die entsprechend des Detailgrades spezifiziert sind. Dabei werden einerseits die Abbildbarkeit auf die Informationsstruktur (siehe Abbildung 25), andererseits die Lesbarkeit und die Flexibilität zur Beschreibung der Benutzerschnittstelle berücksichtigt.

### 5.5.1 Das abstrakte Benutzerschnittstellendesign

In der Spezifikationsphase werden weder die grafische Ausgestaltung noch die Elemente einer 3D-Benutzerschnittstelle betrachtet. Daher wird nun in einem ersten Konkretisierungsschritt innerhalb der Modellierung festgelegt, welche Elemente die Benutzerschnittstelle bilden, wie viele Elemente dies sind und wie diese gruppiert werden. Der Schritt erfolgt in einer Modellbeschreibung, die aufzeigt, mit welchen Interfaceelementen der Anwender in den interaktiven Prozess tritt. Der Entwurf eines Dialogbildes als abstraktes 3D-Interface erfolgt daher auf Basis von Elementen, die einerseits Interface-, andererseits Gruppierungselemente darstellen, um die bildgestützte Interaktionsstruktur der Benutzerschnittstelle zu beschreiben.

Für den Schritt des abstrakten Interfaceentwurfs wird als Modellierungsgrundlage das UML-Profil »MosaIK:Abstrakt« eingeführt. Dieses stellt mithilfe von *Stereotypen* Erweiterungen von bereits in der UML vorhandenen Modellelementen bereit, indem die UML-Stereotype die Modellelemente klassifizieren. Die innerhalb des MosaIK:Abstrakt-Profils eingesetzten Stereotype werden als Erweiterung der UML-Metaklasse *Class* definiert, womit die Bedeutung der Modellelemente im Hinblick auf den Interfaceentwurf spezialisiert und in der Folge für das abstrakte 3D-Interfacedesign eingesetzt werden können. Das MosaIK:Abstrakt-Profil, wie es in Abbildung 27 veranschaulicht ist, beschreibt die verschiedenen <<Dialogbildelemente>> für den Entwurfsprozess. Dabei dient die <<Darstellungsfläche>> einerseits als Grundfläche für das 3D-Interface, andererseits als Gruppierungscontainer für die weiteren Dialogbildelemente. Auf dieser Grundfläche können Interfacebestandteile platziert werden. In der Folge ist die Verortung unterschiedlicher <<Sichten>> gegeben. Die Sichten repräsentieren die möglichen Bildräume einer 3D-Benutzerschnittstelle. Das MosaIK:Abstrakt-Profil definiert ferner Dialogbildelemente, die einen Interaktionscharakter – ein <<Verhalten>> – aufweisen, wodurch die Veränderlichkeit eines 3D-Interface beschrieben werden kann. Ein Spezialfall eines Verhaltens ist die <<Kopplung>>, die Dialogbildelemente und deren Eigenschaften miteinander verbindet. Dadurch kann die Veränderung von Bildräumen und das elementübergreifende Verhalten von Sichten modelliert werden.

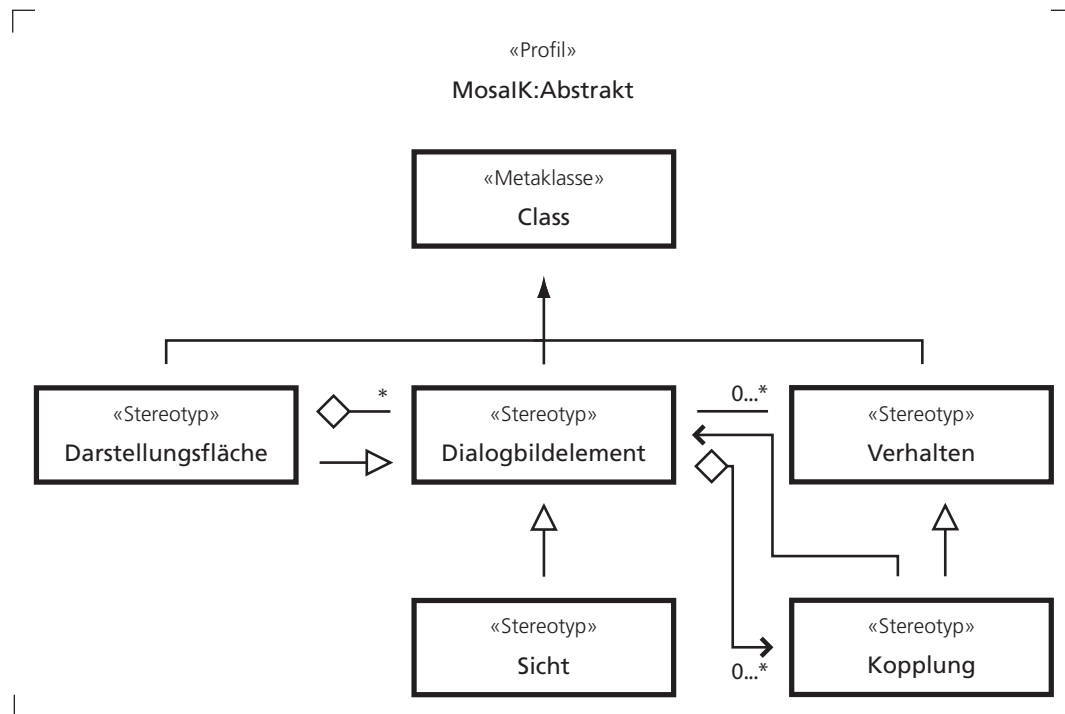


Abbildung 27: Das MosalK:Abstrakt-Profil für den abstrakten 3D-Interfaceentwurf

Aufbauend auf dem UML-Profil und der Anforderungsspezifikation kann nunmehr das abstrakte 3D-Interfacedesign realisiert werden. Die Systematisierung von Darstellungsweisen im dritten Kapitel dieser Arbeit (siehe Abschnitt 3.5) bietet hierfür den nötigen pragmatischen Charakter, um in der Phase der abstrakten Modellierung einbezogen zu werden. Darüber hinaus erfolgt die Festlegung der Dialogstruktur einzelner Dialogbildelemente sowie in Bezug auf deren gegenseitige Verknüpfungen. Dabei bleibt die Modellierung unabhängig von einer computergrafischen Konkretisierung. Das erzeugende Interfacemodell dient als eine allgemeine aber eindeutige Beschreibung der 3D-Benutzerschnittstelle. Die Modellierung eines abstrakten Interfacedesigns wird darüber hinaus durch die im nachfolgenden vorgestellten Entwurfsmuster unterstützt. Hierdurch wird einerseits eine Lösungsvorlage für den visuellen Aufbau der Benutzerschnittstelle zur Verfügung gestellt, andererseits ein Muster für den interaktiven Prozess zwischen Anwender und System aufgezeigt. Das Entwurfsmuster *Dialoggestalter* dient der Erzeugung von Sichten, durch welche die aufgezeigten Visualisierungsziele in den Entwurfsprozess integriert werden können. Das Entwurfsmuster *Regisseur* bietet hingegen eine Vorlage für die Konzeption des Verhaltens eines interaktiven Systems, indem eine Mustervorlage für die Entwicklung eines bildgestützten Interaktionsdesigns für 3D-Benutzerschnittstellen beschrieben wird.

### 5.5.2 Entwurfsmuster: Dialoggestalter

Im Nutzer-Bild-Dialog dienen veränderliche Darstellungen eines dreidimensionalen Objektraums dem interaktiven Prozess zwischen Anwender und System. In Abhängigkeit von den Nutzungsanforderungen sowie dem Nutzer und dessen Aufgabe variieren die bildstrukturellen Charakteristiken des Dialogbildes. Gleichzeitig bietet die Darstellungsfläche der

späteren interaktiven 3D-Anwendung einen „Rahmen“ für die unterschiedlichen Bildräume. Ein singulärer Bildraum erreicht dabei die maximale Ausnutzung der verfügbaren Darstellungsfläche, während mehrere Sichten die Darstellungsfläche aufteilen (siehe Unterabschnitt 3.4.3). Durch den interaktiven Charakter einer 3D-Benutzerschnittstelle ist eine einmalige Festlegung der Anzahl und der Ausgestaltung von Sichten unzureichend hinsichtlich des Nutzer-Bild-Dialoges. Notwendigerweise unterliegt eine 3D-Benutzerschnittstelle im interaktiven Prozess einer ständigen Gestaltsänderung. Diese wird herbeigeführt, indem eine nutzer- oder systemseitige Änderung der Benutzerschnittstelle vollzogen wird. Dabei ist für die Umsetzung von gegebenen Anforderungen der Einsatz von linearen und nichtlinearen Darstellungsformen notwendig (siehe Unterabschnitt 3.4.2), um gegebene Visualisierungsziele (siehe Unterabschnitt 3.5.1) zu erreichen. Das Identifizieren und Konzipieren von grafischen Repräsentationen aus gegebenen Nutzungsanforderungen ist ein wesentlicher Schritt für die Umsetzung des 3D-Interface. Die Bestimmung der bildstrukturellen Eigenschaften des jeweiligen Bildraums und die Festlegung der Sichtenanzahl sind jedoch nicht trivial. Vielmehr können der Aufbau sowie die Anzahl von Sichten durch einen inadäquaten Einsatz den Nutzer-Bild-Dialog stören und die Darbietung von Informationen negativ beeinflussen. Wenn der Entwurf einer interaktiven 3D-Benutzerschnittstelle unter Berücksichtigung von konkreten Visualisierungszielen notwendig ist und gleichzeitig eine Flexibilität und Erweiterbarkeit den interaktiven Nutzer-Bild-Dialog gewährleisten soll, bietet das Entwurfsmuster »*Dialoggestalter*« eine Lösungsvorlage.

Mit dem Dialoggestalter wird der Entwurf von Bildräumen und deren zielgerichteter Zusammenführung auf einer Darstellungsfläche beschrieben. Dies erfolgt, indem einerseits eine Festlegung für einzelne Darstellungen stattfindet, andererseits dieser Vorgang in Abhängigkeit von Eigenschaften und Relationen zu anderen Sichten vollzogen wird. Bei der Ausgestaltung eines Dialogbildes sollte das Visualisierungsziel im Vordergrund stehen, infolgedessen Sichten spezifiziert und zusammengefasst werden, die zur Erreichung dieses Visualisierungszieles notwendig sind. Die Abbildung 28 veranschaulicht das Prinzip, das durch den Dialoggestalter auf Basis des MosalK:Abstrakt-Profils beschrieben wird. Das Entwurfsmuster ermöglicht die Modellierung von 3D-Benutzerschnittstellen in Abhängigkeit von einem *Visualisierungsziel*, indem ein Interface als eine anforderungsspezifische Benutzerschnittstellenausprägung – eine *Dialoggestalt* – beschrieben wird. Das Element *Interface* repräsentiert die Darstellungsfläche einer Benutzerschnittstelle und bildet jeweils die Basis für die Sichten. Wie in der grafischen Form veranschaulicht, ist die Anzahl der Sichten dabei nicht begrenzt, wobei jede Sicht ein Bild des zugrunde liegenden Objektraums repräsentiert. Der *Bildraum*, der für jede Sicht individuell festzulegen ist, wird im Hinblick auf ein situatives Visualisierungsziel spezifiziert. Hierzu wird auf das identifizierte und klassifizierte Repertoire an Darstellungsweisen zurückgegriffen, die im Element der *Darstellungsform* subsumiert sind. Auf Basis des Fundus an bestehenden Visualisierungsverfahren erfolgt die Festlegung der Interfacestruktur auf der Darstellungsfläche. Eine solche Interfacekonfiguration beschreibt zusammenfassend die Dialoggestalt als eine definierte Ausprägung eines Dialogbildes, das in der späteren Anwendung eingesetzt werden kann, um dem Anwender das Erreichen der verschiedenen Visualisierungsziele zu ermöglichen.



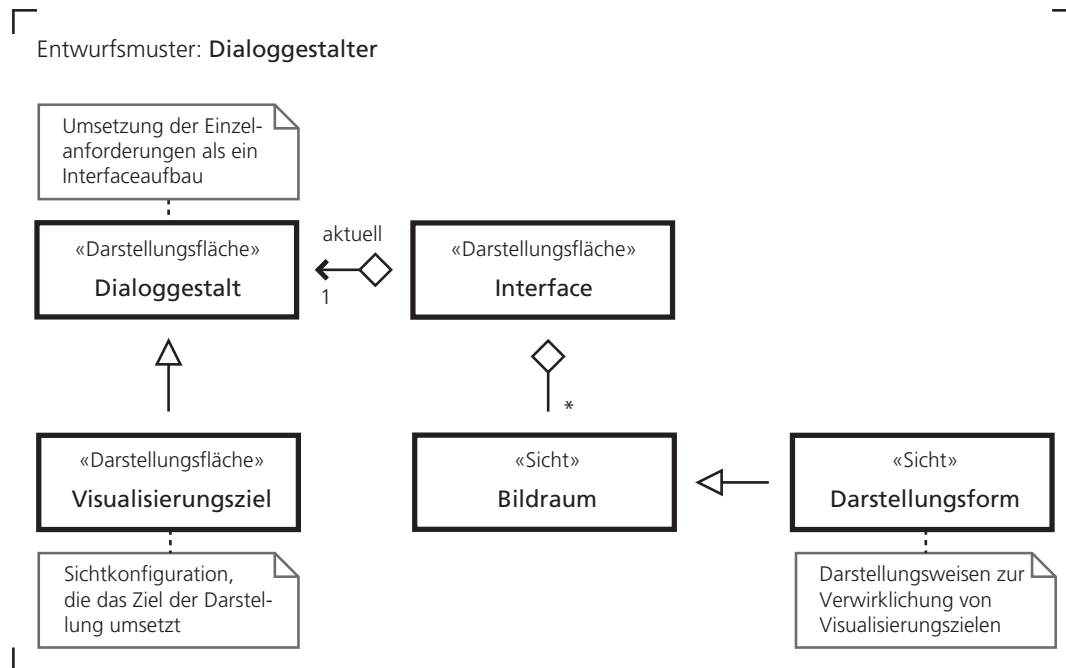


Abbildung 28: Das Entwurfsmuster Dialoggestalter (abstraktes Interfacedesign) beschreibt den Benutzerschnittstellenentwurf von interaktiven Mehrsichten-Interfaces unter Einbeziehung linearer und nichtlinearer Darstellungsverfahren

Auf Basis des Dialoggestalters können Festlegungen für das Interface in Abhängigkeit von einem konkreten Visualisierungsziel getroffen werden. Dadurch kann bereits im abstrakten Entwurf das Zusammenwirken von Bildräumen konzipiert und ferner Inkonsistenzen in der Gestaltung verhindert werden. Weiterhin ist durch den Entwurf von Einzelsichten mithilfe des Dialoggestalters eine Erweiterbarkeit und Flexibilität gewährleistet, welche einerseits die Integration von Darstellungsformen, andererseits von Sichtkonfigurationen umfasst. Darüber hinaus dient die Beschreibung einer Benutzerschnittstelle auf Basis dieses Entwurfsmusters als eine abstrakte Umsetzung eines 3D-Interface für eine frühzeitige Validierung im Anwendungskontext. Bei der Anwendung des Entwurfsmusters ist es zweckmäßig, Richtlinien und Hinweise zur Gestaltung des Nutzer-Bild-Dialoges (siehe Abschnitt 3.4) in die Modellierung einfließen zu lassen. Dadurch wird die Einbindung von unkonventionellen Darstellungsformen, die bisher aufgrund mangelnder Kenntnis keinen Einsatz im Interfaceentwurf finden, unterstützt. Durch die Einbindung einer zusätzlichen Abstraktionsebene in Form der Dialoggestalt führt der Gebrauch des Entwurfsmusters jedoch bei Interfaceentwürfen mit einem singulären Bildraum beziehungsweise mit nur einem expliziten Visualisierungsziel zu einem erhöhten Entwurfsaufwand. Außerdem bietet das Entwurfsmuster keine inhärenten Hinweise für den zweckmäßigen Einsatz von Darstellungsformen in einer konkreten Dialoggestalt, weshalb das Muster vor allem in Verbindung mit den Gestaltungshinweisen und der Systematisierung von Visualisierungszielen eingesetzt werden sollte.

Eine Anwendung des Entwurfsmusters ist beispielsweise im Bildsprache LiveLab verwirklicht, um Benutzerschnittstellen für interaktive Systeme im Bereich der Verkehrsüberwachung zu konzipieren (vgl. [STARKE U. A. 2011, S. 4–5]). Die Realisierung interaktiver Benutzerschnittstellen unter Anwendung des Prinzips des Dialoggestalters geschieht ferner in der Anwendung

*GeoWizard* von FELDT U. A. für die Visualisierung von statistischen Daten in spezifischen Bezugsräumen (vgl. [FELDT U. A. 2005]). Der Anwender wird bei der Exploration statistischer Daten durch gekoppelte Sichten in unterschiedlichen Anordnungen in dessen Dialog mit der Anwendung unterstützt (vgl. [FELDT U. A. 2005, S. 139–140]). In [KEEFE U. A. 2009] ist die beschriebene Mustervorlage hingegen zur Darstellung biomechanischer Bewegungsdaten verwirklicht, um eine Verknüpfung räumlicher und zeitlicher Daten für wissenschaftliche Analysen bereitzustellen. Dabei wird der Anwender in Abhängigkeit von seinen Zielen und Aufgaben durch die Anpassung der Benutzerschnittstelle in seinem bildgestützten Interaktionsprozess geführt. Für den Entwurf von Handlungs- und Verhaltensabläufen im interaktiven Prozess, unter Einbeziehung der unterschiedlichen Ausprägungen eines Dialogbildes, dient das Entwurfsmuster *Regisseur*, welches nachfolgend beschrieben wird.

### 5.5.3 Entwurfsmuster: Regisseur

Der interaktive Prozess zwischen Nutzer und System bewirkt eine fortwährende Abwandlung des Erscheinungsbildes einer 3D-Benutzerschnittstelle, indem sowohl Anpassungen an der Bildstruktur als auch am Verhalten erfolgen. Der Entwurf der dafür notwendigen Dialogtechniken sollte die Etablierung einer funktionalen und dialogorientierten Kommunikation zum Ziel haben. Dabei müssen einerseits die Techniken für den Dialog mit einer Sicht, andererseits die Auswirkungen dieses Dialoges auf die weiteren Sichten der Benutzerschnittstelle modelliert werden. Der Dialog kann im Kontext interaktiver 3D-Darstellungen durch direkte Veränderungen (beispielsweise die Einnahme einer neuen Perspektive in einer Sicht) oder auf Basis kontinuierlicher Veränderungen (beispielsweise ein Hineinzoomen in einen Objektraum) ausgestaltet werden. Weiterhin können direkte und kontinuierliche Dialogtechniken zu komplexen Verhaltensweisen für den bildgestützten Interaktionsprozess kombiniert werden. Durch die verfügbaren Freiheitsgrade innerhalb einer 3D-Sicht einerseits und durch die Verknüpfungsvarianten von Sichten, basierend auf den Sichteigenschaften, andererseits, besteht eine Vielzahl von Interaktionsmöglichkeiten, die im Entwurfsprozess eingesetzt werden können. Ein unstrukturiertes Vorgehen bei der Festlegung des Verhaltens eines 3D-Interface kann zu einem intransparenten und inkonsistenten Interaktionsdesign führen. So kann beispielsweise die Festlegung von Sichtenkopplungen zu einem widersprüchlichen Interaktionsdesign führen, wenn Verknüpfungen als transitive Abhängigkeiten definiert werden. Hierdurch wird die Nachvollziehbarkeit des Interfaceverhaltens für den Anwender beeinträchtigt. Ebenso kann die Definition von Sichtenkopplungen, ohne eine Berücksichtigung des sichtinternen Verhaltens, Inkonsistenzen im Dialogablauf der Benutzerschnittstelle zur Folge haben.

Das Entwurfsmuster »*Regisseur*« bietet eine Vorlage für die Modellierung von Dialogtechniken für 3D-Benutzerschnittstellen mit koordinierten multiplen Sichten. Um in einem Interfaceentwurf, welcher auf der Festlegung einer Menge von Einzeltechniken beruht, die Konsistenz des Interaktionsdesigns und die Nachvollziehbarkeit der Abhängigkeiten sicherzustellen, sollte die Konzeption ausgehend von der einzelnen Sicht erfolgen. Ein definierter Interaktionsumfang für eine jede Sicht kann durch zusätzliche Sichtenkopplungen ergänzt werden, um deren Verhalten zu erweitern. Die Abbildung 29 zeigt das Entwurfsmuster in schematischer

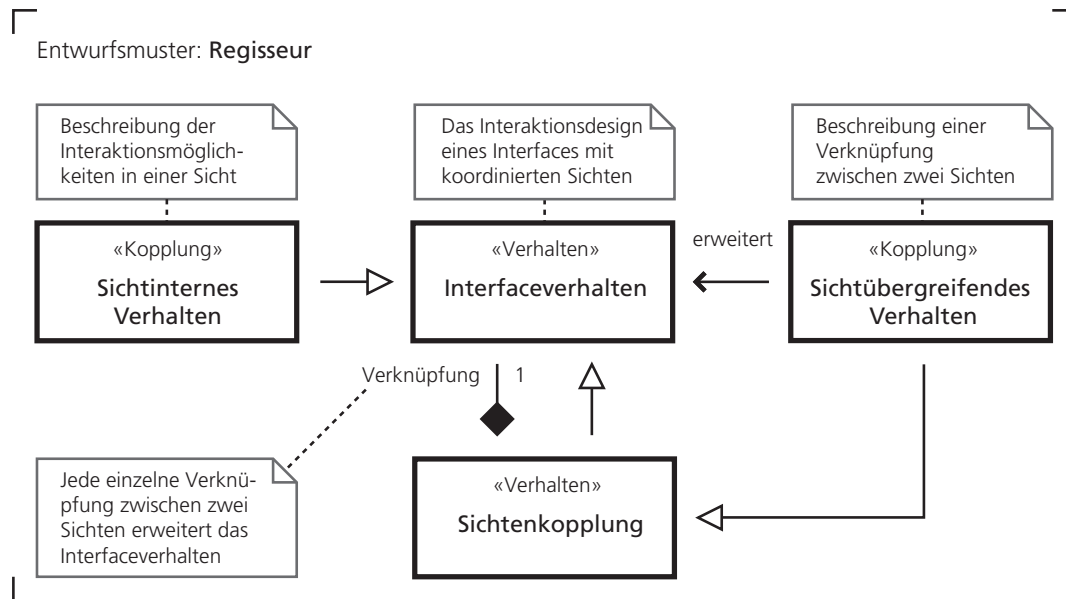


Abbildung 29: Das Entwurfsmuster **Regisseur** (abstraktes Interfacedesign) bietet eine Vorlage für die Modellierung eines Interfaceverhaltens, das durch eine Festlegung von sichtinternen und sichtübergreifenden Interfaceveränderungen beschrieben ist.

Form. Das *Interfaceverhalten* ist zunächst durch die Festlegung des *sichtinternen Verhaltens* beschrieben. Hierbei kann es sich um Dialogtechniken für die vollständige Bewegungsfreiheit in einer Sicht oder um ein eingeschränktes Interaktionsdesign handeln. Das Verhalten der Sichten sollte im Entwurfsprozess jeweils kontextabhängig und sichtspezifisch festgelegt werden. Das definierte Verhalten der Sichten dient für den weiteren Entwurfsprozess als Grundkonfiguration. Aufbauend auf dem sichtinternen Verhalten können nunmehr Abhängigkeiten als Erweiterungen des Verhaltens von Sichten definiert werden. Eine *Sichtenkopplung* dient dazu das Verhalten einer Sicht dynamisch zu verändern, indem durch das *sichtübergreifende Verhalten* auf die Veränderungen in einer zweiten Sicht Bezug genommen wird. Durch die Festlegung mehrerer Sichtenkopplungen können infolgedessen einfache und zusammengesetzte sichtübergreifende Verhaltensweisen definiert werden. Da ein sichtinternes Verhalten und eine Kopplung weiterhin ein Interfaceverhalten beschreiben, ist es möglich, dass eine Kopplung sowohl in Abhängigkeit zu einem sichtinternen Verhalten als auch zu einem sichtübergreifenden Verhalten gesetzt werden kann. Eine flexible Kombination von Sichtverhalten und Sichtenkopplungen ist dadurch möglich.

Im Zusammenwirken mit dem Entwurfsmuster Dialoggestalter bietet der Regisseur eine Vorlage für die abstrakte Beschreibung des interaktiven Kommunikationsprozesses zwischen Nutzer und System. Auf Basis dieses Entwurfsmusters sind Dialogtechniken zur Beschreibung koordinierter multipler Sichten flexibel erstellbar und wiederverwendbar. Gleichzeitig können inkonsistente Interaktionsdesigns vermieden werden, da jede Sichtenkopplung lediglich eine Abhängigkeit beschreibt und die Kopplungen rekursiv beschrieben sind. Die Schachtelung von Dialogtechniken kann jedoch durch eine große Anzahl von Kopplungen die Übersichtlichkeit des Interaktionsdesigns beeinträchtigen. Außerdem ist die Identifikation von kontextspezifischen Unzulänglichkeiten in komplexen Entwürfen aufgrund der zahlreichen Abhängigkeiten erschwert.

Das Funktionsprinzip des Regisseurs findet beispielsweise in 3D-Modellierungswerkzeugen wie Autodesk Maya [AUTODESK] Anwendung, um für die parallelprojektiven und perspektivischen Sichten individuelle Dialogtechniken bereitzustellen. Gleichzeitig erfolgt eine sichtübergreifende Kopplung, indem ausgewählte Objekte in allen Sichten farblich hervorgehoben werden. In der Entwicklungsumgebung Bildsprache LiveLab wiederum erfolgt eine Nutzung des Entwurfsmusters, indem für aufgabenbezogene Navigations- und Interaktionsmechanismen komplexe Sichtenkopplungen definiert werden können. Mit diesen werden koordinierte multiple Sichten zur Untersuchung von Objektsymmetrien und für den Vergleich von Objektverhältnissen umgesetzt (vgl. [STARKE U. A. 2011, S. 72–73]).

### 5.5.4 Das konkrete Benutzerschnittstellendesign

Während im Schritt der abstrakten Modellierung die Vorgehensweise des Anwenders im Nutzer-Bild-Dialog unter Berücksichtigung eines notwendigen Gestaltungsfreiraums fokussiert wird, erschwert dieser die Entwicklung von präzisen 3D-Benutzerschnittstellen. Die Ursache dessen ist, dass ein abstraktes Interfacedesign die tatsächliche Ausgestaltung einer Benutzerschnittstelle nicht beschreibt. Infolgedessen besitzt der abstrakte Entwurf weder Angaben zur verfahrenstechnischen Erzeugung der computergrafischen Bildräume noch zum parametrisierten Interaktionsdesign oder dem exakten strukturellen Aufbau. Daher schließt sich innerhalb der Modellierungsphase nunmehr der konkrete 3D-Interfaceentwurf an. Dabei werden die konzeptionellen, strukturellen und gestalterischen Beschreibungen sowie Entwürfe im Hinblick auf eine Realisierung als benutzer- und anwendungsgerechte 3D-Benutzerschnittstelle konkretisiert.

Mit der Modellierung einer konkreten 3D-Benutzerschnittstelle soll der erforderliche Detailgrad erreicht werden, der eine Überführung in eine plattformspezifische Beschreibung ermöglicht. Dazu werden für ein konkretes 3D-Interfacedesign die Elemente und deren Relationen auf Basis eines UML-Profiles festgelegt. Das Profil ist an den Detailgrad der Entwurfsphase angepasst und konkretisiert darüber hinaus das UML-Profil für das abstrakte 3D-Interfacedesign. Das »MosaIK:Konkret« Profil ist in Abbildung 30 dargestellt und zeigt die Interfaceelemente als UML-Stereotype und deren Zusammenhänge auf Basis der UML-Syntax. Ferner ist gezeigt, wie die Elemente des abstrakten Profils, unter Berücksichtigung von deren Assoziationen, in die konkrete Beschreibung übersetzt sind. Die Abbildung 30 zeigt, dass die Darstellungsfläche durch den Interfacebereich, die Sicht durch das Panel, die Kopplung durch den Operator und das Verhalten durch den Modifikator im MosaIK:Konkret-Profil konkretisiert werden und dadurch als Stereotype für die gegenständliche Modellierung zur Verfügung stehen. Das Profil enthält weiterhin eine Menge von Stereotype, die als Erweiterung der UML-Metaklasse *Class* definiert sind.

Das Element *Interfacebereich* bildet die Darstellungsfläche der abstrakten Beschreibung ab und repräsentiert die Grundfläche einer 3D-Benutzerschnittstelle auf der eine beliebige Anzahl von *Panels* platziert werden können. Gleichzeitig stellt der Interfacebereich selbst ein Panelement dar, das wie jedes Panel über ein *Raster* verfügt. Dieses Raster fungiert als Grundlinienraster für die Anordnung der Panels. Dadurch kann einerseits die Grundfläche

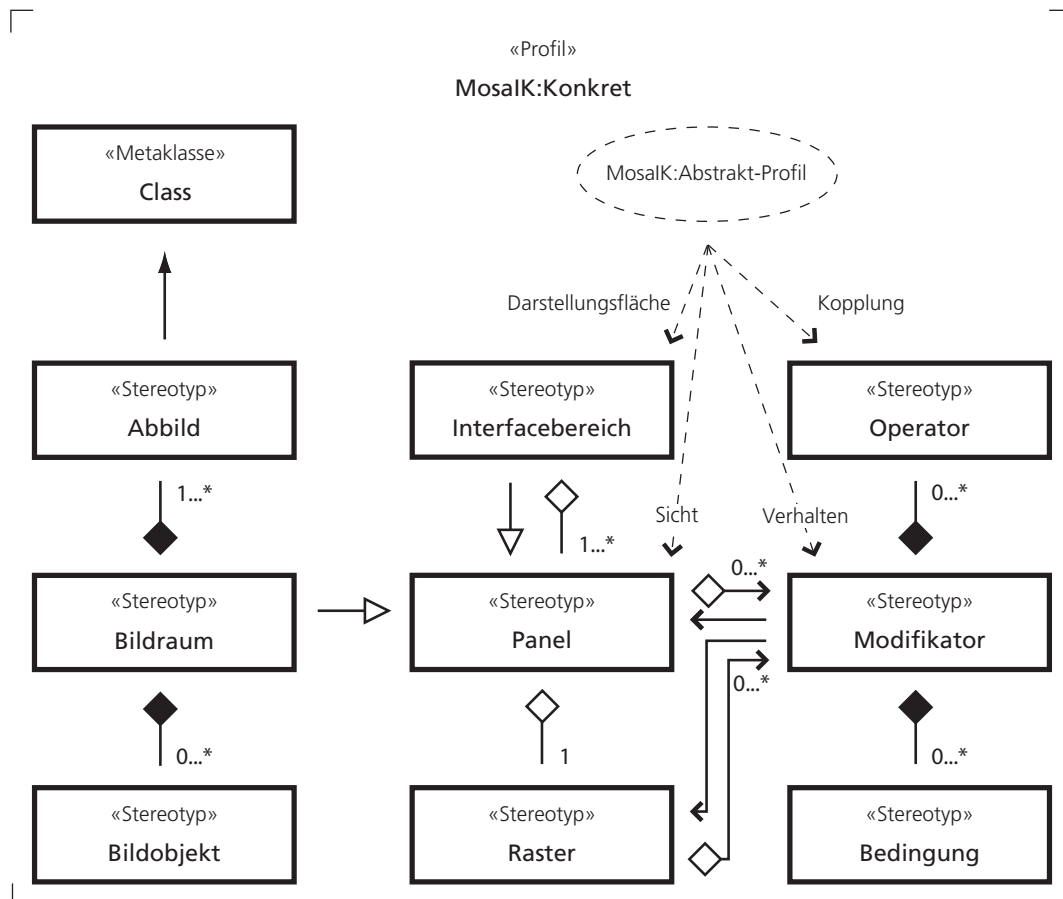


Abbildung 30: Das MosalK:Konkret-Profil für den konkreten 3D-Interfaceentwurf präzisiert die Elemente des MosalK:Abstrakt-Profiles unter der Berücksichtigung semantischer Zusammenhänge

einer Benutzerschnittstelle zur Anordnung von Panels gerastert werden, andererseits wird jedes Panel zur Platzierung von Bildräumen mit einem Raster versehen. In der Ausgestaltung kann darüber hinaus eine computergrafische Visualisierung in das Panel integriert sein und dadurch einen *Bildraum* beschreiben, der auf Basis einer Anzahl von *Abbildern* und der Verfügbarkeit von *Bildobjekten* erzeugt wird. Weiterhin zeigt sich eine Konkretisierung des Verhaltens im MosalK:Konkret-Profil durch das Element *Modifikator*, das sich aus einer Anzahl von *Operatoren* zusammensetzt, wobei diese auf Basis von *Bedingungen* beeinflusst und gesteuert werden können. Ein Operator legt eine diskrete oder eine kontinuierliche Veränderung einer Paneleigenschaft beziehungsweise eines Attributes des Bildraums oder des Rasters fest.

Übereinstimmend mit dem abstrakten Interfaceentwurf werden zur Unterstützung des Modellierungsprozesses Entwurfsmuster bereitgestellt. Die Entwurfsmuster *Raster* und *Sichtweise* bieten hierzu Vorlagen für die Strukturierung und Anordnung von Sichten sowie für die Konzeption von flexiblen und erweiterbaren Kameramodellen. Das Entwurfsmuster *Wandler* zeigt eine Möglichkeit den visuellen Strukturen dynamisch und kontextabhängig Darstellungseigenschaften zuzuweisen, während auf Basis des Entwurfsmusters *Modifikation* das konkrete Verhalten einer interaktiven 3D-Anwendung definiert werden kann.

## Entwurfsmuster: Sichtweise

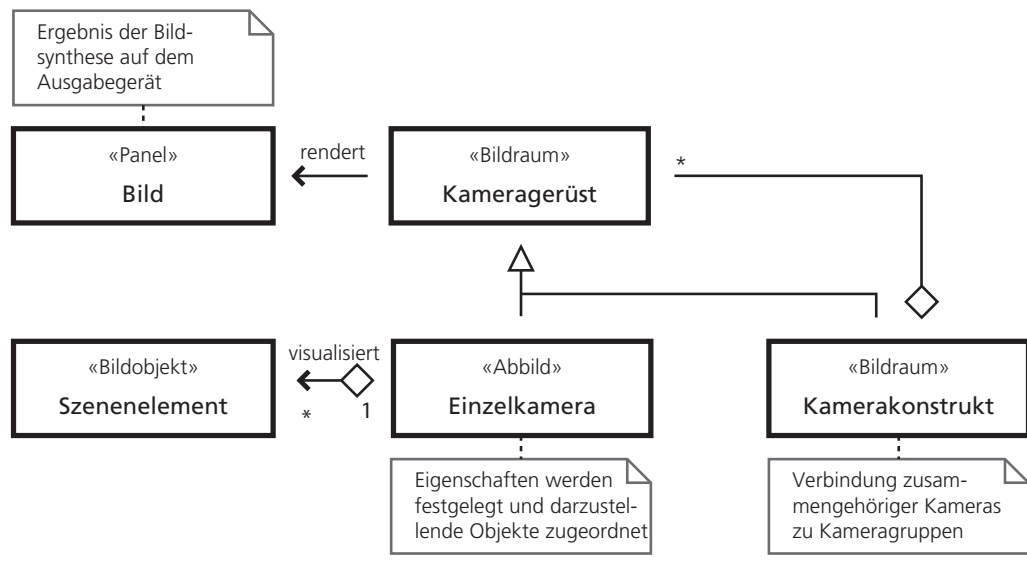


Abbildung 31: Das Entwurfsmuster Sichtweise (konkretes Interfacedesign) beschreibt die Strukturierung von Kameras in Form eines Teil-Ganzes-Hierarchieprinzips zur Erstellung komplexer Kameramodelle

### 5.5.5 Entwurfsmuster: Sichtweise

Im Unterabschnitt 3.1.2 wird aufgezeigt, dass in computergrafischen Anwendungen die Realisierung einer 3D-Benutzerschnittstelle fast ausschließlich unter Einbeziehung zentral- und parallelprojektiver Bildräume erfolgt. Zur Verwirklichung einer Reihe von Visualisierungszielen (siehe Abschnitt 3.5), ist diese Beschränkung auf lineare Projektionsverfahren nicht opportun. Krummlinige und multiperspektivische Darstellungen erreichen situativ einen zusätzlichen Nutzen für die Visualisierung dreidimensionaler Objekträume. Die Konzeption von Kameramodellen zur Erzeugung krummliniger und multiperspektivischer Darstellungen ist jedoch aufgrund der Anzahl an Freiheitsgraden einer computergrafischen Kamera und der Menge an manipulierbaren Kameraparametern weder für einen Designer noch für einen Entwickler eine triviale Aufgabe.

Das Entwurfsmuster »Sichtweise« bietet eine Lösungsvorlage für die Konstruktion von Kameramodellen. Mit dem Entwurfsmuster ist es möglich, computergrafische Kameramodelle für lineare, kurvilineare und multiprojektive Abbildungsverfahren zu entwerfen. Da sich komplexe Kameramodelle durch eine Kombination von computergrafischen Einzelkameras erstellen lassen, beschreibt das Entwurfsmuster die Umsetzung eines Teil-Ganzes-Hierarchieprinzips, wie es in der Softwareentwicklung durch das Kompositum Entwurfsmusters in einem vergleichbaren Problemkontext verwendet wird (vgl. [GAMMA u. A. 1994, S. 239 ff.]). Mithilfe der Sichtweise werden baumartige Strukturen erzeugt, deren Blätter die computergrafischen Kameras bilden (siehe Abbildung 31). Jede Kamera wird durch ein lineares oder ein nichtlineares Projektionsverfahren definiert, das in der praktischen Umsetzung durch eine

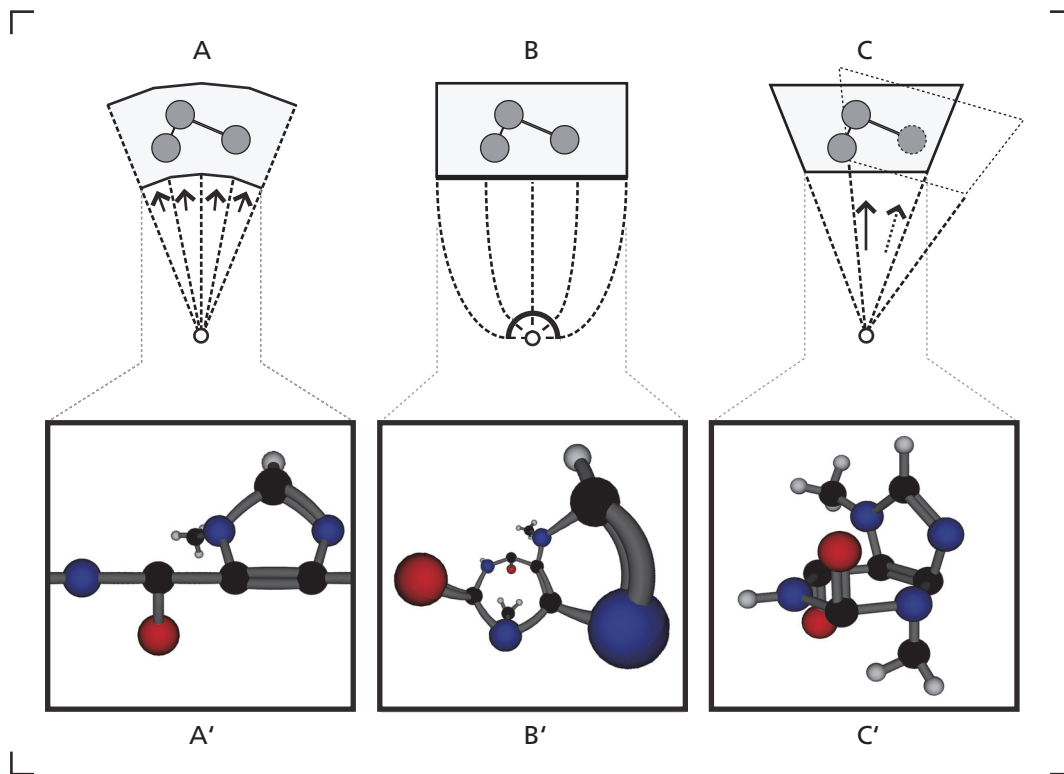


Abbildung 32: Beispiele von Kameramodellen auf Basis des Entwurfsmusters Sichtweise und resultierende computergrafische Darstellungen eines fiktiven Moleküls (bildraum-basierte Multiperspektive (A), krummlinige Perspektive (B), objektraum-basierte Multiperspektive (C)) (Quelle: Bildsprache LiveLab [@AG-TV])

Modifikation des computergrafischen Kameramodells erreicht wird (siehe Unterabschnitt 3.3.2). Ferner werden jeder Kamera die darzustellenden *Szenenelemente* zugewiesen, wobei dies einerseits die gesamte visuelle Struktur sein kann, andererseits nur einzelne Objekte umfasst. Die *Einzelkameras* können aufgrund des Teil-Ganzes-Hierarchieprinzips des Musters durch eine Komposition in *Kameragerüsten* konstruiert werden. Hierdurch wird die modellhafte Erstellung von objektraumbasierten und bildraum-basierten Multiprojektionen umsetzbar. Die unterschiedlichen Kameragerüste sind wiederum kombinierbar und können in der Folge ein komplexes *Kamerakonstrukt* bilden. Auf Basis des entwickelten Kameramodells erfolgt die computergrafische Bildsynthese zur Erstellung eines *Bildes* in einer 3D-Benutzerschnittstelle.

Durch den hierarchischen Aufbau, der dem Sichtweise-Entwurfsmuster zugrunde liegt, kann ein komplexes Kamerakonstrukt, bestehend aus unterschiedlichen Kameragerüsten und Einzelkameras, einheitlich behandelt werden. Dabei ist die Anwendung des Entwurfsmusters unabhängig von der Anzahl der zu verwendenden Einzelkameras. Ferner ist durch den strukturellen Aufbau die Flexibilität und die Erweiterbarkeit von Kameramodellen gewährleistet, da zusätzliche Kameragerüste einem Konstrukt angefügt werden können. In der Folge besteht die Möglichkeit existierende Kameramodelle im Entwurfsprozess wiederzuverwenden. Das Entwurfsmuster der Sichtweise ermöglicht Designern komplexe und gleichzeitig strukturierte Kameramodelle zu definieren, ohne eine detaillierte Kenntnis über die computergrafische Bildsynthese besitzen zu müssen. Für den Softwareentwickler bietet die Anwendung des



Entwurfsmusters die Möglichkeit, computergrafische Kameras, vergleichbar mit grafischen Primitiven zur Erstellung komplexerer 3D-Modelle, für einen flexiblen und gleichzeitig modularen Aufbau von heterogenen Kameramodellen einzusetzen. Inwieweit die Abhängigkeiten der Kameras im Kamerakonstrukt oder in den Kameragerüsten zu definieren sind, wird jedoch nicht durch das Entwurfsmuster vorgegeben. Vielmehr erfolgen die Entwurfsentscheidungen durch den Designer oder den Entwickler in Abhängigkeit von den Nutzungsanforderungen und dem Anwendungskontext. Die Sichtweise ermöglicht, dass beispielsweise Fischaugendarstellungen mit einer Einzelkamera (siehe Abbildung 32-B) ebenso wie multiperspektivische Darstellungen auf Basis einer Vielzahl von Kameras (siehe Abbildung 32-A und Abbildung 32-C) spezifiziert werden können.

Die Anwendbarkeit des Entwurfsmusters wird durch interaktive Systeme demonstriert, die das Prinzip der Sichtweise einsetzen. Die Applikation von Yu u. A. nutzt die Methode für multiperspektivische Panoramadarstellungen auf Basis von linearen Einzelkameras (vgl. [Yu & McMillan 2004]), während POPESCU u. A. strukturierte Kameramodelle für die Auflösung von Verdeckungen in räumlichen Szenen einsetzen (vgl. [POPESCU u. A. 2009]). Die Anwendung des Entwurfsmusters kann jedoch durch den vergleichsweise einfachen Gestaltungsprozess von Kameramodellen zur Verwendung einer großen Anzahl Kameras führen, wodurch Softwareentwickler vor die Herausforderung gestellt werden, die Echtzeitfähigkeit eines interaktiven Systems sicherzustellen. Lediglich das gezielte Einsetzen von multiprojektiven und nicht-linearen Darstellungsweisen (siehe Gestaltungshinweis in Abschnitt 3.4.2) kann diesem Problem teilweise entgegenwirken.

### 5.5.6 Entwurfsmuster: Panelraster

Eine konkrete Nutzeraufgabe sowie ein gegebener Datenbestand können den Einsatz multipler Sichten bedingen, um den Interaktionsaufwand für die Informationsdarbietung sowie den kognitiven Aufwand für die Erfassung der Informationen zu reduzieren (vgl. [WANG BALDONADO u. A. 2000; NORTH & SHNEIDERMAN 1997]). Die Gestaltungshinweise in Abschnitt 3.4.3 verdeutlichen, dass die Position, die Größe und die Anordnung von Sichten wesentliche Kriterien in Bezug auf die Unterstützung der visuellen Kommunikation auf planaren Darstellungsflächen sind. In den aktuellen, interaktiven 3D-Anwendungen sind nur wenige vordefinierte Anordnungen bereitgestellt oder die Anordnung von Sichten wird dem Anwender übergeben, der lediglich durch eine systemseitige Platzierungsfunktionalität in der Strukturierung von Sichten unterstützt wird. Diese unzureichende beziehungsweise nutzerseitige Sichtenstrukturierung steht jedoch im Widerspruch zur Relevanz einer Sichtenanordnung in Bezug auf die bildgestützte Interaktion zwischen Nutzer und System. Der strukturelle Aufbau sollte bereits im Entwurfsprozess konzipiert und festgelegt werden, da durch die Formation der Sichten die Aufmerksamkeit des Anwenders situativ und kontextbezogen gelenkt werden kann (vgl. [EISNER 2001]). Da sich jedoch im Unterschied zu konventionellen 2D-Benutzeroberflächen die Gestalt des Dialogbildes durch den interaktiven Charakter der Bildräume unentwegt verändert, muss die Anordnung von Sichten weitgehend ohne Kenntnis über die konkrete Ausgestaltung des Bildraums erfolgen. Daher ist insbesondere bei Benutzerschnittstellen mit einer großen Anzahl an Sichten beziehungsweise einer hohen

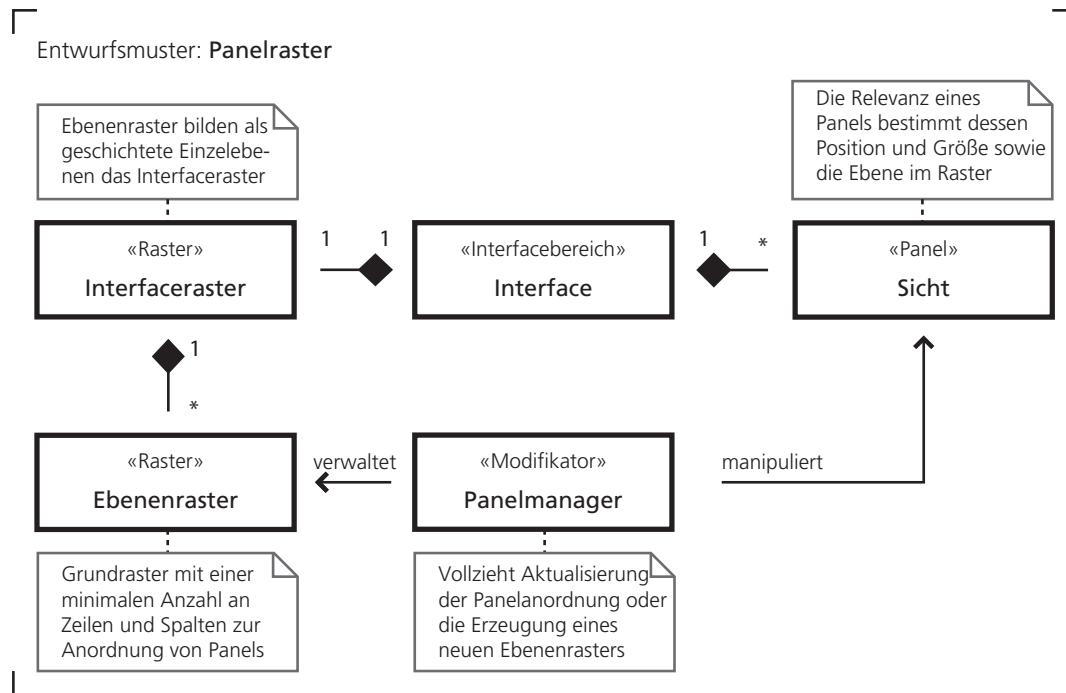


Abbildung 33: Das Entwurfsmuster **Panelraster** (konkretes Interfacedesign) bietet eine Vorlage für den Aufbau und die Anordnung von Sichten auf Basis von Grundlinienrastern

Varianz – in Bezug auf die Positionierung, die Anordnung und die Größe von Sichten – der strukturierte Aufbau einer Benutzerschnittstelle, unter Einbeziehung der Gestaltungshinweise (siehe Unterabschnitt 3.4.3), wesentlich.

Eine Vorlage für einen strukturellen Aufbau von 3D-Benutzerschnittstellen wird mit dem Entwurfsmuster »*Panelraster*« gegeben, das in schematischer Form in Abbildung 33 dargestellt wird. Der Entwurf zur Anordnung multipler Sichten ist hierbei an den Umgang mit Gestaltungsrastern im Druck- und Webseitendesign angelehnt und für die 3D-Interfacegestaltung adaptiert. Generell dienen Gestaltungs raster der Organisation und der Ordnung einer zunächst leeren Darstellungsfläche. Mit einem Raster als Grundlage wird die Aufgabe vereinfacht, die zu visualisierenden Interfaceelemente anzuordnen, um in der Folge visuell konsistente Benutzerschnittstellen zu gestalten. Für eine Einbindung von Ordnungsprinzipien in den Entwurfsprozess von 3D-Interfaces setzt sich das Panelraster aus drei wesentlichen Aspekten zusammen: dem *Interface*, dem *Interfaceraaster* und den *Panels*. Das Interfaceraaster legt die Vorgaben des strukturellen Aufbaus der Benutzerschnittstelle fest. Dieses kann aus mehreren *Ebenenrastern* bestehen, die in geschichteter Form angeordnet werden und zusammen das Interfaceraaster bilden. Ein Ebenenraster wiederum beschreibt ein planares Grundlinienraster durch die Festlegung von Zeilen und Spalten. Auf diesen Ebenenrastern werden Panels platziert. Diese sind Platzhalter für die Sichten der 3D-Benutzerschnittstelle, wie diese beispielsweise auf Basis des Entwurfsmusters Sichtweise erzeugt werden können, und beschreiben in der Folge eine konkrete Hilfskonstruktion zur Organisation der grafischen Elemente auf den Ebenenrastern. Jedes Panel wird durch die Angabe der Position und die horizontale sowie vertikale Ausdehnung eindeutig auf einem Ebenenraster platziert. Durch

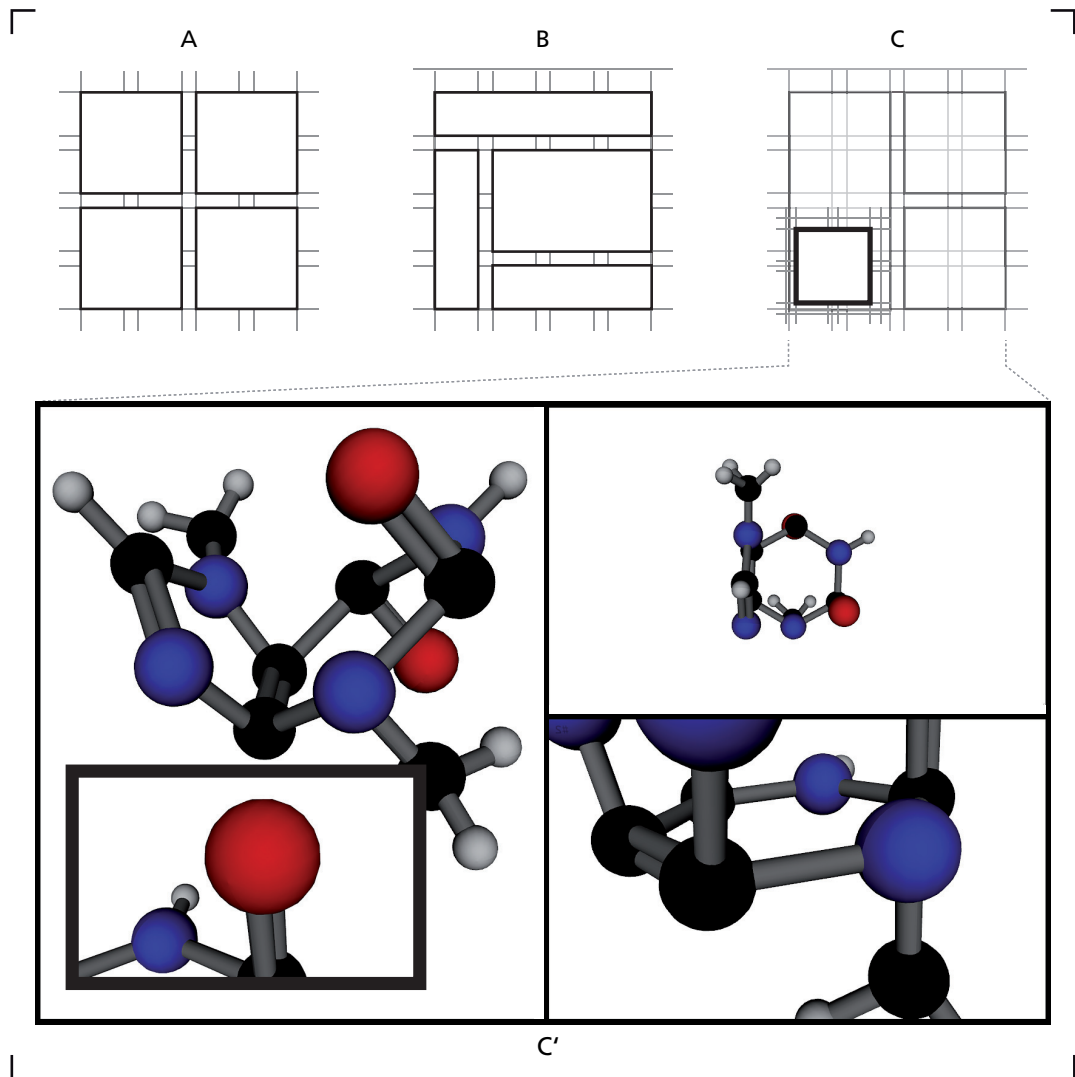


Abbildung 34: Beispiele für Sichtenanordnungen mithilfe der Mustervorlage Panelraster (A-C) sowie eine 3D-Benutzerschnittstelle bestehend aus multiplen Sichten, die auf zwei übereinanderliegenden Ebenenrastern (C') angeordnet sind (Quelle: Bildsprache LiveLab [AG-TV]).

die Möglichkeit einer Schichtung von Ebenenrastern und deren Platzierung auf dem Interface, können Interfaceraaster erstellt werden, die zu einer teilweisen beziehungsweise vollständigen Verdeckung von Panels eines tieferen Ebenenrasters führen. In der Folge können umschließende Sichtenanordnungen oder in Teilen überlappende Formationen verwirklicht werden.

Für die Sicherstellung einer konsistenten Anordnung der Panels auf einem Ebenenraster müssen die Lage- und Größenänderungen dieser in Relation zueinander erfolgen. Das Konstrukt des *Panelmanagers* dient im Entwurfsmuster einer Konsistenzwahrung auf den Ebenenrastern. Dies erfolgt, indem Veränderungen in der Form, Größe und Anzahl der Panels einerseits durch die Ausdehnung des Ebenenrasters begrenzt sind, andererseits Überdeckungen von Panels auf einem Ebenenraster ausgeschlossen werden. Die Ausgestaltung des Grundlinienrasters sowie die Panelanordnungen, die für eine interaktive Benutzerschnittstelle

benötigt werden, können somit unter Berücksichtigung der Schnittstellenveränderlichkeit festgelegt werden. Die Festlegung der tatsächlichen Veränderungen und Übergänge des strukturellen Aufbaus im interaktiven Prozess zwischen Nutzer und System werden ferner durch die Entwurfsmuster *Wandler* und *Modifikation* konkretisiert, die in den nachfolgenden Unterabschnitten vorgestellt werden.

Das Entwurfsmuster *Panelraster* zeigt eine Lösung für die Integration einer strukturellen Ordnung unter Beibehaltung der Flexibilität in einer 3D-Benutzerschnittstelle, ohne deren interaktiven Charakter einzuschränken. Das *Panelraster* begrenzt bei dessen Anwendung weder den Austausch noch die Erweiterbarkeit von Interfacerastrern, da Ebenenraster erstellt und individuell definiert aber auch entfernt werden können. Mithilfe des Entwurfsmusters können Designer und Entwickler existierende Regeln und Hinweise aus den Bereichen der Print-, Comic- und der Webseitengestaltung in den Entwurfsprozess von 3D-Benutzerschnittstellen integrieren. Beispielsweise kann auf Basis des *Panelrasters* die Forderung VAN WELIES in [VAN WELIE & TRAETTEBERG 2000, S. 6] nach einer einheitlichen horizontalen und vertikalen Unterteilung eines Interfacerastrers zur Anordnung grafischer Elemente in Benutzerschnittstellen verwirklicht werden (siehe Raster A in Abbildung 34). Ebenso kann, wie Raster B in Abbildung 34 zeigt, die Aufteilung von Panels unter Berücksichtigung der Leserichtung erfolgen (vgl. [EISNER 2001]), während durch die Panelanordnung auf dem Raster C in Abbildung 34 die Hervorhebung einer expliziten Sicht erreicht wird, indem das priorisierte Panel in einem vordergründigen Ebenenraster verortet ist. Die exemplarischen *Panelraster* zeigen jedoch ebenso, dass eine freie Justierung von Panels außerhalb des Definitionsbereiches, aufgrund der vorgegebenen Einordnung in ein definiertes Raster, nicht gegeben ist. Außerdem führt das Einbinden von Panels in ein Ordnungssystem dazu, dass Veränderungen der Größe und Gestalt eines Panels zu einer Neujustierung der zu platzierenden Bildräume führt und unter Voraussetzung eines komplexen Schnittstellenaufbaus eine erhebliche Umstrukturierung der Dialogbildelemente verursachen kann. Daher ist der Entwurf eines konkreten Interfacedesigns durch ein enges Zusammenwirken der Entwurfsmuster *Sichtweise* und *Panelraster* geprägt, wodurch der strukturelle Aufbau einer 3D-Benutzerschnittstelle unter Berücksichtigung der im Panel zu platzierenden Sichten realisiert werden kann. Die Verwendung von Interfacerastrern erfolgt im wissenschaftlichen Bereich, unter anderem in CAD-Systemen, um die oftmals multidimensionalen Daten in einer kontextbezogenen Darstellungsweise zu arrangieren. Dazu werden die Daten abgesehen von einer singulären, perspektivischen Sicht in einer parallelprojektiven Darstellung als eine Zwei-Fenster-Ansicht präsentiert oder in einer Vier-Fenster-Ansicht für die Darstellung des Objektraums aus unterschiedlichen Perspektiven dargeboten. Weiterhin werden überlappende und einschließende *Panelraster* im Unterhaltungsbereich (vgl. [GLOBAL SOUTHERN]) sowie im Lehr- und Lernbereich zur Einblendung temporärer Informationen, als zusätzliche Ebene, eingesetzt.

### 5.5.7 Entwurfsmuster: Wandler

Interaktive 3D-Anwendungen bieten die Möglichkeit, die Darstellungsweise des Objektraums während der Laufzeit zu ändern. Der bildgestützte Kommunikationsprozess, wie dieser in Abschnitt 2.2.2 beschrieben ist, wird nutzerseitig mit dem Ziel vollzogen, durch die interaktive

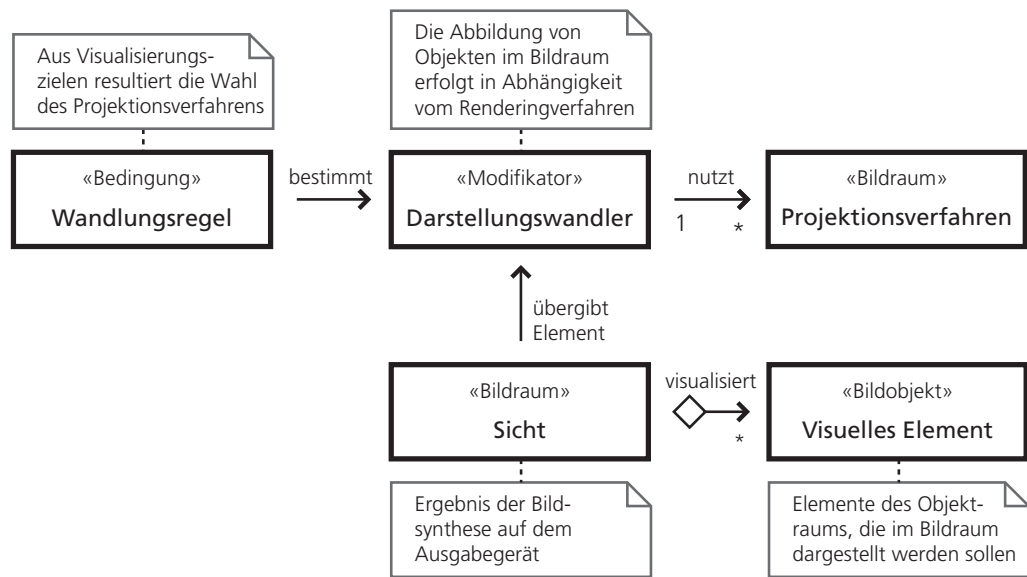
Entwurfsmuster: **Wandler**

Abbildung 35: Das Entwurfsmuster Wandler (konkretes Interfacedesign) bietet eine Vorlage für die Trennung einer visuellen Struktur von der Darstellungsmethode und der regelbasierten dynamischen Zuweisung bei der Bilderzeugung.

Veränderung des Erscheinungsbildes einer visuellen Struktur ein intendiertes Visualisierungsziel zu erreichen. Die systemseitige Veränderung des Bildraums ist dabei notwendig, da auf Basis eines allgemeingültigen Erscheinungsbildes ein effektiver und effizienter Nutzer-Bild-Dialog nicht verwirklicht werden kann. Die bereits aufgezeigte Vielgestaltigkeit des computergrafischen Bildraums (siehe Unterabschnitt 3.5) resultiert im Entwurfsprozess jedoch in der Verfügbarkeit einer großen Anzahl an Darstellungsformen, die im interaktiven Prozess eine situative Anwendung finden können. Diese Vielfalt erschwert die Konzeption einer Benutzerschnittstelle, da die Verwirklichung eines 3D-Interface, unter Berücksichtigung einer ständigen Veränderbarkeit zur Laufzeit der Anwendung, erfolgen muss. Das Entwurfsmuster »Wandler« bietet eine Lösungsvorlage, die notwendige Flexibilität und situationsabhängige Anpassbarkeit des Bildraums in den Entwurfsprozess zu integrieren. Das Muster basiert dazu auf einem Softwaremuster für die Kodierung visueller Variablen für 2D-Informationsvisualisierungen (vgl. [HEER & AGRAWALA 2006, S. 857–858]). Das Entwurfsmuster bietet eine Vorlage mit der *visuelle Elemente* des Objektraums den unterschiedlichen Abbildungsverfahren flexibel zugeordnet werden können (siehe Abbildung 35). Durch die Zuweisung von Abbildungseigenschaften zur Laufzeit einer Anwendung kann in der Folge der *Bildraum* kontextsensitiv an den Nutzer-Bild-Dialog angepasst werden. Die dynamische Zuweisung der visuellen Elemente zu einer Darstellungsweise erfolgt durch dedizierte, wiederverwendbare *Projektionsverfahren*, welche die Eigenschaften der visuellen Elemente auf unterschiedliche Art und Weise aus dem Objektraum in den Bildraum überführen. Die zur Verfügung stehenden Projektionsverfahren werden dazu mithilfe des *Darstellungswandlers* dynamisch den visuellen Elementen zugeordnet. Diese Zuordnung erfolgt durch *Wandlungsregeln*, die

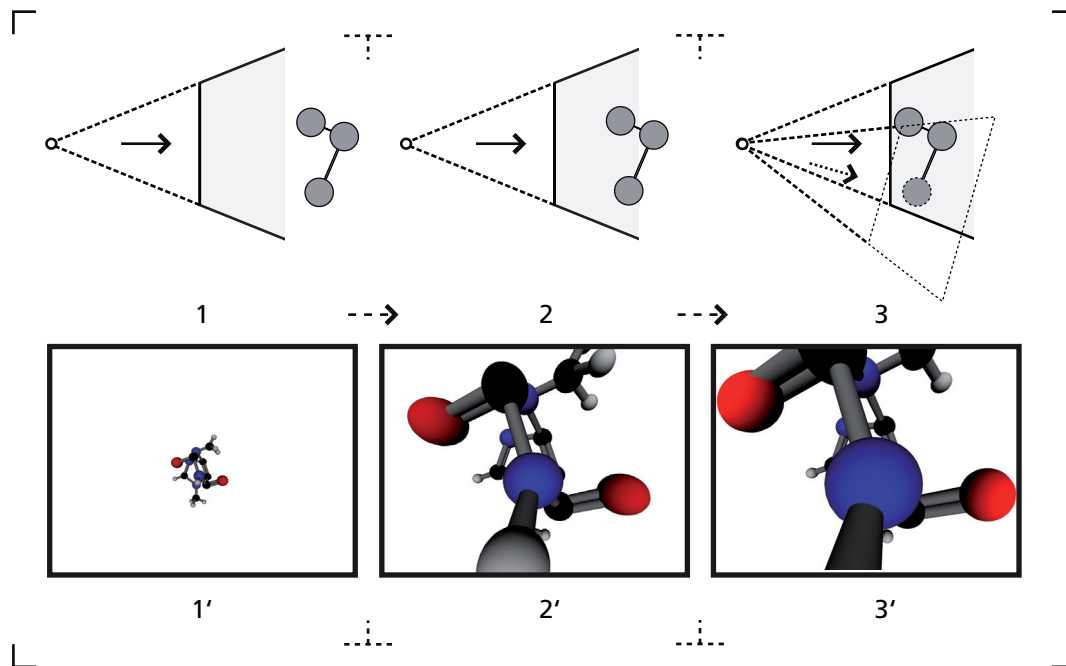


Abbildung 36: Beispiel für die Funktionsweise des Entwurfsmusters Wandler veranschaulicht eine distanzbasierte Veränderung des Bildraums, ausgehend von einer monoperspektivischen zu einer multiperspektivischen Ansicht zur Reduzierung projektionsbedingter Verzerrungen (Quelle: Bildsprache LiveLab [@AG-TV]).

auf Basis der Objektattribute sowie der Objektraum- und Kameraeigenschaften zu definieren sind. Die konkrete visuelle Repräsentation der Elemente kann in der Folge zur Laufzeit einer interaktiven Anwendung auf Basis einer Bibliothek an Darstellungsverfahren und unter Einsatz vordefinierter Bedingungen dynamisch zugewiesen werden.

Durch das Entwurfsmuster Wandler wird ein Funktionsprinzip beschrieben, womit die Elemente eines Objektraums gemeinsam verwaltet werden können, obgleich diese in einem interaktiven System auf Basis unterschiedlicher Darstellungsweisen abgebildet werden. Die Abbildung 36 verdeutlicht die Anwendung des Wandlers anhand des Einsatzes einer objektraumbasierten Multiperspektive zur Erhaltung der Objektgestalt in einer Detailansicht. Dazu werden in Abhängigkeit von der Distanz zwischen der visuellen Struktur und dem computergrafischen Augpunkt unterschiedliche Kameramodelle für die Visualisierung eingesetzt. Die Abbildung zeigt, dass in einer Überblicksdarstellung die visuelle Struktur zunächst monoperspektivisch dargestellt wird (Abbildung 36-1). Mit dem Übergang in eine Detailansicht treten projektionsbedingte Verzerrungen auf (Abbildung 36-2), die durch den Einsatz eines multiprojektiven Abbildungsverfahrens minimiert werden. Dies erfolgt, indem die roten Kugeln des fiktiven Moleküls in Abbildung 36-3 in individuellen Perspektiven dargestellt werden. Diese dynamische Zuweisung des Projektionsverfahrens, durch die Übergabe von visuellen Elementen an den Darstellungswandler, wie es das Beispiel verdeutlicht, kann dabei sowohl nutzer- als auch systembedingt ausgelöst werden. Mit der Anwendung des Wandler-Entwurfsmusters kann die Variabilität eines Dialogbildes zur Laufzeit einer Anwendung bereits im Entwurfsprozess berücksichtigt werden. Gleichzeitig ist eine Flexibilität im Entwurfsprozess gewährleistet, die ferner eine Erweiterbarkeit durch das Hinzufügen neuer Projektionsverfahren zulässt. Weiterhin kann jedoch durch den Einsatz des Wandlers die

Komplexität einer 3D-Anwendung, aufgrund der flexiblen Anpassbarkeit des Bildraums, zunehmen. Bei interaktiven 3D-Systemen, die auf Basis des Entwurfsmusters verwirklicht werden, ist darüber hinaus eine erhöhte Ressourcenauslastung möglich, da die bereitstehenden Abbildungsverfahren zur Laufzeit reaktiviert, adaptiert und ausgeführt werden müssen. Den zusätzlichen Nutzen, der sich durch die Anwendung des Entwurfsmusters für die Gestaltung interaktiver Systeme ergibt, zeigen SUDARSANAM U. A. in [SUDARSANAM U. A. 2008] durch einen situativen Einsatz verschiedener Darstellungsverfahren zur Exploration medizinischer Daten. Weiterhin wird durch SINGH und BALAKRISHNAN das zugrunde liegende Prinzip des Wandlers genutzt, um Bildraumveränderungen, ausgehend von der Aufmerksamkeit des Nutzers, herbeizuführen (vgl. [SINGH & BALAKRISHNAN 2004]). Dabei wird die Anpassung der Visualisierung an das Nutzerverhalten und das Visualisierungsziel des Anwenders erreicht.

### 5.5.8 Entwurfsmuster: Modifikation

In 3D-Anwendungen erfolgt durch den Interaktionsprozess zwischen Nutzer und System eine stetige Veränderung der Gestalt einer Benutzerschnittstelle. Damit einerseits die Aktionen des Anwenders, andererseits die Auswirkungen auf Veränderungen der zugrunde liegenden Daten in einem 3D-Interface verbildlicht werden können, ist die Spezifikation von Dialogtechniken im Entwurfsprozess erforderlich. Unter anderem kann im Nutzer-Bild-Dialog eine direkte Veränderung einer Kameraeigenschaft – beispielsweise die Neuausrichtung im Raum – vom Anwender gefordert sein. Ebenso kann eine kontinuierliche Modifikation einer Eigenschaft innerhalb eines festgelegten Zeitraumes, zum Beispiel eine Kamerafahrt zur Umsetzung einer Zoom-Interaktion, ein Erfordernis darstellen. Weiterhin kann es wesentlich sein, verschiedene Veränderungsoperationen zu komplexen Dialogtechniken zu verbinden. Dabei kann, wie in Abbildung 36 dargestellt, mit einer Veränderung der Kameraposition ein Wechsel des Projektionsverfahrens verknüpft sein, wobei der Wechsel des Abbildungsverfahrens in Abhängigkeit von der Entfernung zu den Elementen im Objektraum erfolgt. Die Techniken zur Modifikation einzelner Sichten sowie für deren koordiniertes Zusammenwirken müssen im Entwurfsprozess festgelegt werden. Dazu ist zunächst die Granularität, die einer Modulation des Dialogbildes zugrunde liegt, zu bestimmen, da sich durch diese die Möglichkeiten in der Gestaltung des Nutzer-Bild-Dialoges ergeben. In derzeitigen Anwendungen werden die Interaktionen vorwiegend in konkreter Zuordnung zum Bildraum in monolithischen Einheiten gekapselt. So wird in der Anwendung Autodesk Maya eine orthografische Draufsicht zur Verfügung gestellt, in der ein Anwender ausschließlich Lateral- und Zoombewegungen ausführen kann, während in der perspektivischen Sicht eine freie Navigation verwirklicht ist. Durch eine solche Kapselung von Dialogtechniken wird sowohl in der Konzeption als auch in der Umsetzung von 3D-Interfaces ein Interaktionsrepertoire bereitgestellt, das in ein interaktives System integriert werden kann. Die Erzeugung neuer Techniken erfolgt durch die Abwandlung oder die Spezialisierung existierender Funktionseinheiten oder durch die vollständige Neuentwicklung einer Dialogtechnik und deren Aufnahme in die Menge existierender Interaktionsformen. Durch den Einsatz bestehender originärer oder lediglich modifizierter Funktionseinheiten wird jedoch der Designer in der Gestaltung von Benutzerschnittstellen eingeschränkt. Eine vollständige Neuentwicklung von Dialogtechniken



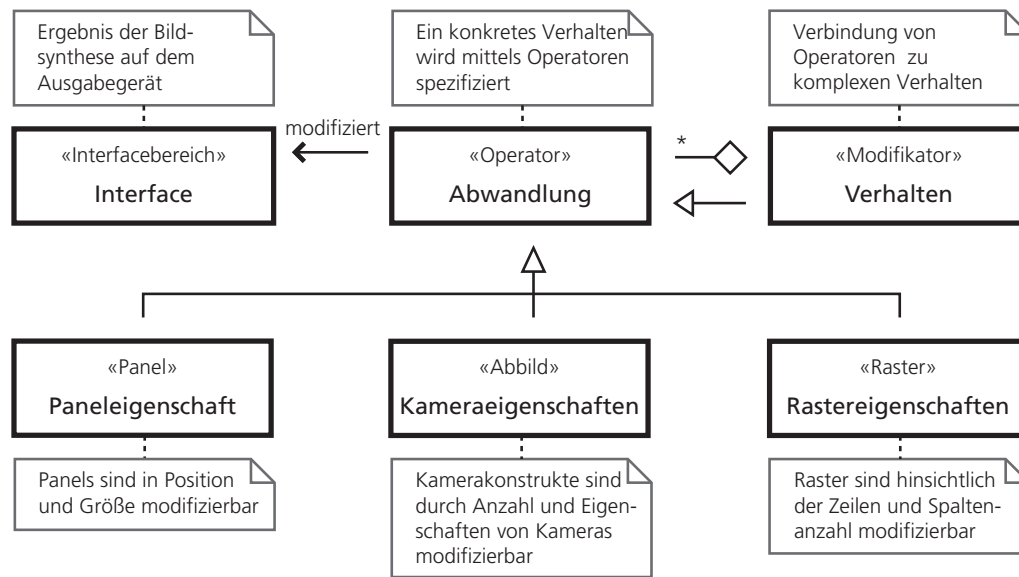
Entwurfsmuster: **Modifikation**

Abbildung 37: Das Entwurfsmuster **Modifikation** (konkretes Interfacedesign) bietet eine Vorlage zur Erstellung von Sichtenkopplungen in einer 3D-Benutzerschnittstelle durch die Komposition von Einzeloperatoren.

hat wiederum für den Entwickler einen erheblichen Aufwand in der Umsetzung zur Folge. In der Konsequenz werden überwiegend etablierte Dialogtechniken in interaktiven Anwendungen eingesetzt.

Ein alternativer Ansatz wird mit dem Entwurfsmuster »*Modifikation*« beschrieben, das eine Vorlage für den Entwurf von modularisierbaren Dialogtechniken bietet. Das Entwurfsmuster beschreibt hierbei den Entwurf von Techniken basierend auf einem feineren Granularitätsgrad als es bestehende Interaktionsformen bieten. Dazu werden zunächst mögliche Dialogtechniken in deren veränderliche Eigenschaften zerlegt. In der Folge können die Parameter von *Kamera*, *Panel* und *Raster* individuell spezifiziert und deren Veränderlichkeit als kombinierbare Operatoren (*Abwandlungen*) zusammengefügt werden (siehe Abbildung 37). Ein Operator beschreibt in der Folge eine Modulation des Bildraums beziehungsweise des Sichtenaufbaus einer 3D-Benutzerschnittstelle. Die Operatoren können, wie Abbildung 37 zeigt, wiederum durch entwurfsspezifische Gruppierung zu komplexen Modifikatoren (*Verhalten*) kombiniert werden, die beispielsweise durch den Nutzer als Navigation durch den Raum eingesetzt oder durch die stetige Verfolgung eines animierten Objektes durch die Kamera für den Anwender sichtbar wird. Darüber hinaus können Operatoren in Relation zu anderen Operatoren gesetzt werden. Dadurch werden die Modulationen in eine relative oder absolute Abhängigkeit zueinander gesetzt, infolgedessen eine Kopplung von Eigenschaften der Benutzerschnittstelle erreicht wird. Dies wird in Abbildung 38 durch die Koordination zweier Sichten mit einander entgegengesetzten Kamerabewegungen verdeutlicht, indem durch die Kopplung ein gleichzeitiges Hinein- und Hinauszoomen verwirklicht ist.

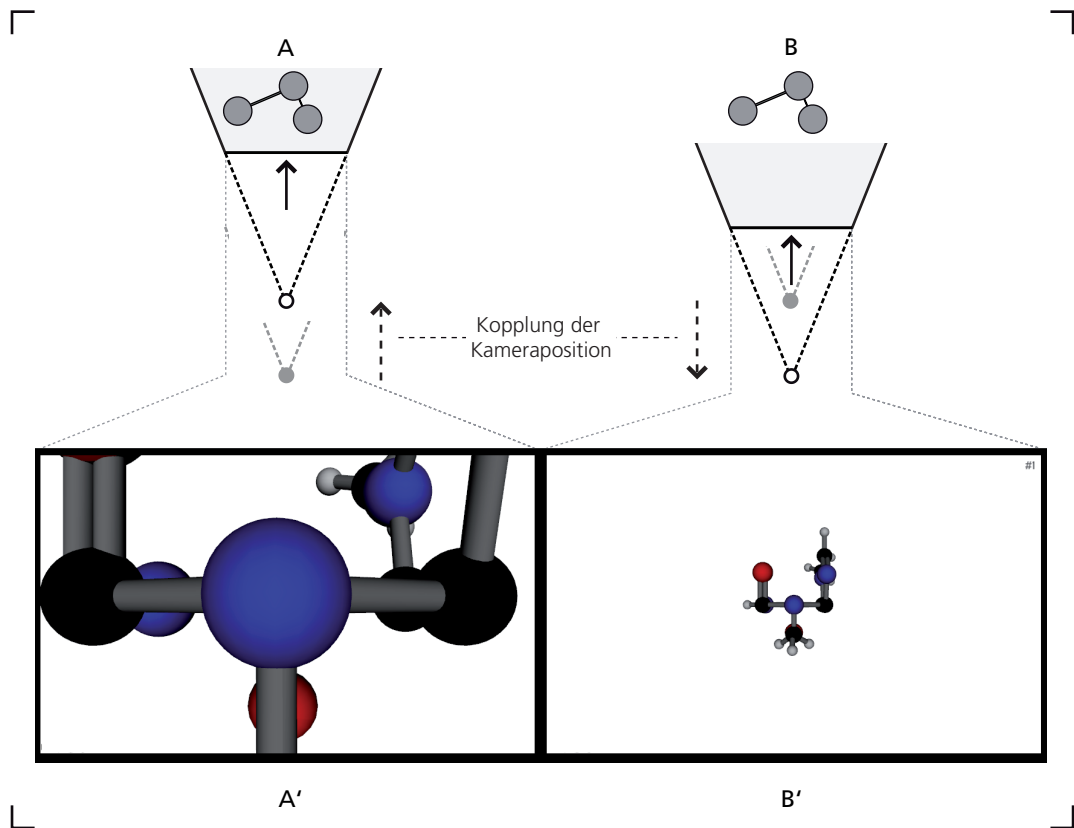


Abbildung 38: Beispiel einer Sichtenkopplung mithilfe des Entwurfsmusters Modifikation zur Umsetzung eines Zoom-In-Out Verhaltens einer 3D-Benutzerschnittstelle (Quelle: Bildsprache LiveLab [AG-TV]).

Das Entwurfsmuster Modifikation bietet zusammenfassend die Möglichkeit ein komplexes Verhalten in einer flexiblen Form zu gestalten, wodurch bildstrukturelle Veränderungen in einer Sicht konzipiert und ferner der Entwurf koordinierter multipler Sichten verwirklicht werden können. Durch diesen Aufbau ist ein hohes Maß an Flexibilität hinsichtlich der Konfigurierbarkeit von Dialogtechniken gegeben. Dies schließt eine Erweiterung eines Modifikators durch die Integration weiterer Operatoren ein. Weiterhin ist die Wiederverwendung von bestehenden Modifikatoren nicht ausgeschlossen. Das Interfacelayout (Unterstützung durch das *Panelraster*), die visuelle Codierung von Elementen des Objektraums (auf Basis des *Wandlers*) und die Projektionsverfahren (basierend auf der *Sichtweise*) können in der Folge mithilfe des Modifikation-Entwurfsmusters manipuliert werden. Der gebrauchstaugliche Aufbau eines Modifikators und insbesondere die kontextbezogene Konzeption von Operatoren und deren Zusammenwirken werden jedoch nicht durch das Entwurfsmuster, sondern vielmehr durch die Designer und Entwickler in der Verwendung des Musters erreicht. In der Folge können Inkonsistenzen im Verhalten einer Benutzerschnittstelle nur durch die am Entwurfsprozess beteiligten Personen erkannt und korrigiert werden. Dabei müssen die Anforderungen, die durch den Anwendungskontext bestehen identifiziert werden, um die korrekte Funktionsweise eines Modifikators in einer interaktiven Anwendung sicherzustellen.

Eine Anwendung des Prinzips, das im Entwurfsmuster Modifikation beschrieben ist, dient im *Visualization Toolkit* dazu, dem Anwender flexible Interaktionsmöglichkeiten zur Exploration

von abstrakten Daten bereitzustellen (vgl. [SCHROEDER & LORENSON 1996]). In der Anwendung Bildsprache LiveLab dient das Entwurfsmuster hingegen als Vorlage für die Komposition von Einzeloperatoren zur Spezifikation von Kopplungen in koordinierten multiplen Sichten (vgl. [STARKE U. A. 2011]).

## 5.6 Die Implementierungsphase

Das Modell eines konkreten 3D-Interfacedesigns als das Resultat aus der Modellierungsphase bildet die Grundlage für die Beschreibung einer 3D-Benutzerschnittstelle in der Implementierungsphase. Im Gegensatz zu einer technologie- oder anwendungsgebundenen Entwicklung, beispielsweise auf Basis grafischer GUI-Builder und der teilautomatischen Überführung eines Interfaceentwurfs in Quellcode, bietet die Beschreibung einer Benutzerschnittstelle in einem weitestgehend unabhängigen Format verschiedene Vorteile. Mithilfe einer deklarativen Beschreibungssprache ist es möglich, die Sprache zur Festlegung des Aufbaus und der Gestaltung eines 3D-Interface unabhängig von einer Zielsprache zu spezifizieren. Dadurch kann die Wiederverwendbarkeit von Interfaceelementen verbessert, eine Flexibilität in der visuellen Ausgestaltung erreicht und die Plattformunabhängigkeit der Interfacebeschreibung gewährleistet werden. Hierzu sind durch Beschreibungssprachen einerseits die Elemente eines Interface definiert, andererseits die Beziehung in denen die einzelnen Bausteine zueinander stehen festgelegt. Mit der Beschreibung einer 3D-Benutzerschnittstelle mithilfe einer deklarativen Auszeichnungssprache ist einerseits verbunden, dem Entwickler die softwaretechnologische Realisierung eines 3D-Interface zu erleichtern, andererseits den Designern die Möglichkeit zu geben, Benutzerschnittstellen zu entwerfen, zu beschreiben und ebenso zu nutzen.

Die Beschreibung einer 3D-Benutzerschnittstelle, basierend auf dem Modell des konkreten 3D-Interfacedesigns, umfasst die Spezifikation von Bildräumen und deren Komposition als koordinierte multiple Sichten auf einer Darstellungsfläche. Die bestehenden Auszeichnungssprachen besitzen jedoch einen Fokus auf die Beschreibung des zugrunde liegenden Objektraumes. Die gegebenen Möglichkeiten zur Definition eines Dialogbildes, auf Basis einer bestehenden Beschreibungssprache, sind für die Gestaltung koordinierter multipler Sichten in einer interaktiven 3D-Anwendung, unter Verwendung linearer und nichtlinearer Darstellungsverfahren, infolgedessen nicht ausreichend. Eine resultierende Bestrebung zur Entwicklung eines gänzlich eigenständigen Formates ist nicht zielführend. Dies würde zu einer Neukonzeption von Elementen bestehender Sprachen führen – beispielsweise die Beschreibung von Objekten, Texturen und Lichtquellen – und hätte einen unverhältnismäßigen Entwicklungsaufwand zur Folge. Deshalb wird für die Beschreibung eines Interaktionsbildes für 3D-Anwendungen der Sprachumfang von bestehenden Szenenbeschreibungssprachen, die eine Spezifikation des Datenbildes sicherstellen, um Elemente zur Beschreibung des Dialogbildes ergänzt.

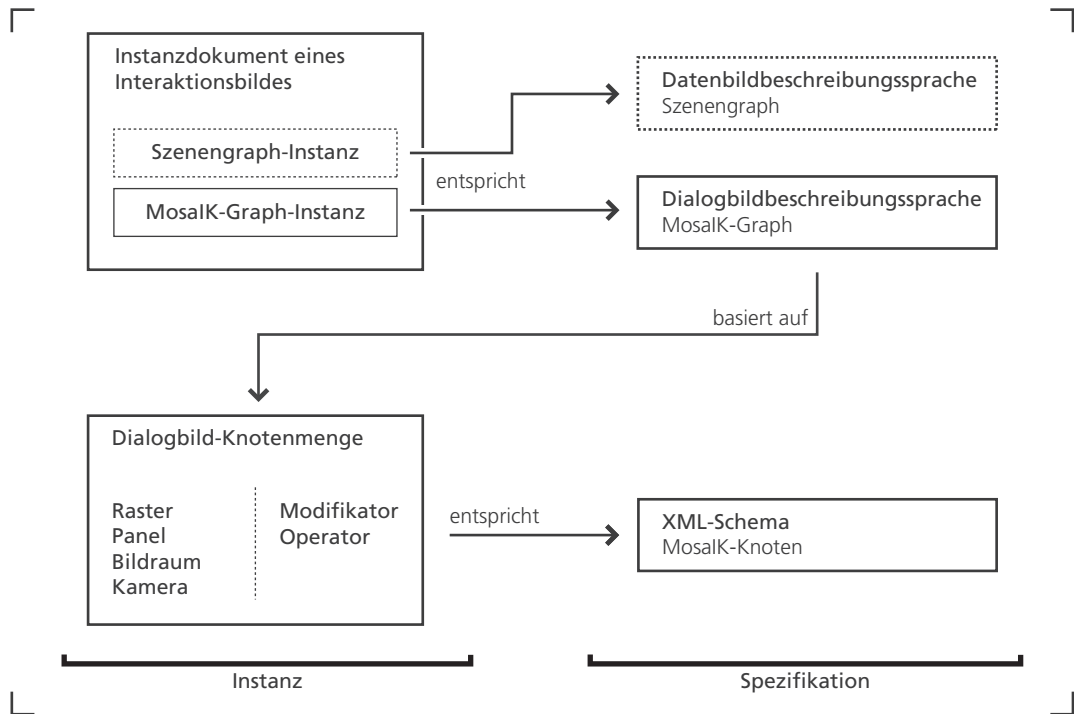


Abbildung 39: Schematische Darstellung der Konzeption des MosaIK:Dialogs für die Beschreibung von Dialogbildern.

### 5.6.1 Konzeption der Beschreibungssprache

Die Implementierung eines Dialogbildes in der Programmiersprache eines interaktiven Systems hat eine bereits identifizierte mangelnde Plattformunabhängigkeit und eine fehlende Wiederverwendbarkeit des Dialogbildes zur Folge. Im Gegensatz dazu unterliegt die Festlegung der visuellen Strukturen, basierend auf Szenenbeschreibungssprachen, dieser Einschränkung nicht. Um ein möglichst einheitliches und darüber hinaus flexibles Konzept zur Beschreibung von Dialogbildern zu erreichen, wird die deklarative Auszeichnungssprache »MosaIK:Dialog« mit den folgenden Kerneigenschaften entwickelt.

- Eine Erweiterbarkeit und Flexibilität des Ansatzes soll den zunächst experimentellen Charakter der zu erzeugenden Dialogbilder berücksichtigen und darüber hinaus die Erweiterung und Weiterentwicklung der Sprache begünstigen.
- Die Sprache soll unabhängig von proprietären 3D-Formaten sein, um eine weitgehende Überführbarkeit in unterschiedliche interaktive Anwendungen sicherzustellen. Ferner soll die Sprache in ein 3D-Format integrierbar sein oder dieses ergänzen.
- Die Beschreibung von Dialogbildern soll durch vordefinierte Dialogbildknoten in einer graphbasierten Struktur realisiert werden.
- Die Spezifikation von Dialogbildknoten basiert auf dem standardisierten Format XML-Schema.

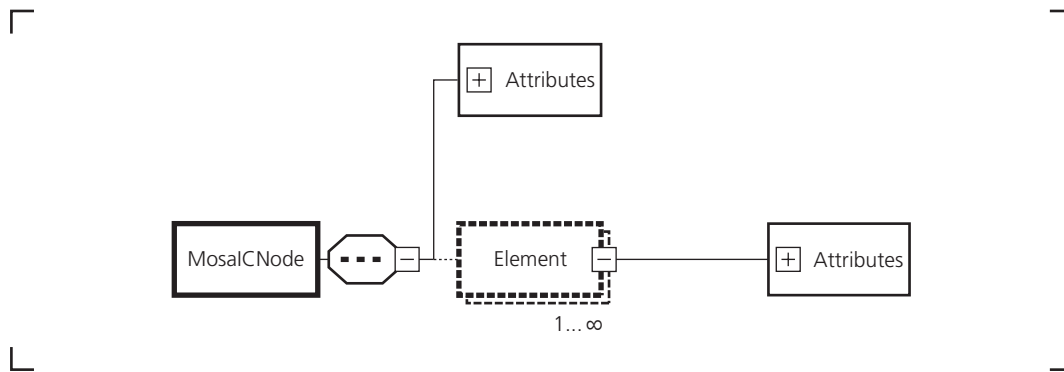


Abbildung 40: Grundlegende Struktur des XML-Schemas MosaiK-Knoten

- Die Darbietung einer 3D-Benutzerschnittstelle auf dem Ausgabegerät soll durch eine Übersetzung von Instanzdokumenten in eine grafische Repräsentation einer interaktiven 3D-Anwendung möglich sein.

Die Abbildung 39 zeigt die Konzeption des Mosaik:Dialog indem die Bestandteile und Zusammenhänge der Beschreibungssprache veranschaulicht werden. Den Ausgangspunkt bildet das XML-Schema »*Mosaik-Knoten*«. Das Schema ist die strukturelle Grundlage für die Definition von Dialogbildknoten. Auf Basis der Spezifikation kann eine Knotenmenge zur Beschreibung eines Dialogbildes entwickelt. Das für die Beschreibung eines Dialogbildes notwendige Knotenrepertoire unterteilt sich in die visuellen Interfaceelemente *Raster*, *Panel*, *Bildraum* und *Kamera*, während durch den *Modifikator* und den *Operator* Verhaltensknoten definiert werden, die zur Beschreibung der Veränderlichkeit des Interface eingesetzt werden. Der Einsatz der Dialogbild-Knotenmenge erfolgt in Form einer definierten Graphenstruktur; dem *Mosaik-Graph*. In der Folge ist die Beschreibung des Dialogbildes einer interaktiven 3D-Anwendung auf Basis einer Knotenmenge in einer Graphenstruktur gegeben und mithilfe von Instanzdokumenten durch eine 3D-Anwendung interpretierbar. Durch die Zusammenführung einer *Instanz des Datenbildes* und einer *Dialogbild-Instanz* ist eine faktische Beschreibung eines *Interaktionsbildes* gegeben. In den nachfolgenden Unterabschnitten werden die verschiedenen Aspekte des Mosaik:Dialog zur Dialogbildbeschreibung erläutert.

### 5.6.2 Knotendefinition für das Dialogbild

Die Grundlage für eine Definition von Knoten zur Beschreibung des Dialogbildes ist das XML-Schema »*Mosaik-Knoten*«, wodurch die Knotenmenge eindeutig spezifizierbar wird. In Abbildung 40 ist das XML-Schema in Diagrammform dargestellt. Die Knotendeklaration umfasst einerseits *Elementdeklarationen*, andererseits *Attributdeklarationen*, wobei jeweils der Name des Elementes oder des Attributs festzulegen ist. Die Attributdeklarationen eines Knotens umfassen Informationen, die den Knoten und dessen Funktionsweise näher beschreiben und dienen in der Folge zur Repräsentation der Funktionalität von Knoten, wodurch die Eigenschaften der Dialogbildelemente definiert werden können. Ein Attribut (*Attribute*) wird zusätzlich zur Angabe eines Bezeichners durch die Festlegung eines Datentyps und eines Standardwertes definiert. Dadurch wird sichergestellt, dass Attribute welche bei der Beschreibung eines Dialogbildes nicht explizit definiert werden, dennoch über einen festgelegten Wert verfügen. Weiterhin ist es in Anlehnung an das DEF/USE Prinzip von X3D

(vgl. [WEB3D]) möglich, einen bereits existierenden Knoten zu referenzieren. Hierdurch wird die Erweiterbarkeit eines MosaIK:Dialog durch die Komposition von Dialogbildknoten gewährleistet. Ein MosaIK-Knoten kann darüber hinaus Elementgruppen (*Element*) zur strukturierten Konfiguration einer Knotenfunktionalität besitzen. Eine Elementgruppe wird, vergleichbar mit der Deklaration von Knotenattributen mittels eines Bezeichners festgelegt, woraufhin wiederum die Attribute deklariert werden können, welche die Elementgruppe enthalten soll.

### 5.6.3 Trennung von Daten- und Dialogbild

In vielen 3D-Beschreibungsformaten erfolgt die Datenhaltung auf Basis eines Szenengraphs, wodurch Objekte und Lichtquellen sowie die weiteren Elemente des Objektraums in einer baumartigen Struktur verwaltet werden (siehe [WEB3D; MICROSOFT]). Durch diese Form der Datenhaltung wird ein struktureller Zusammenhang zwischen der im Objektraum verorteten computergrafischen Kamera und den Szenenobjekten suggeriert, der jedoch in dieser Ausprägung für den Anwender nicht existiert. Zwischen dem Objekt- und dem Bildraum besteht vielmehr ein Übersetzungsverhältnis, das durch die Relation von visueller Struktur und computergrafischer Kamera eine nur vordergründig räumliche Beziehung im Objektraum eingeht. Für die Beschreibung eines Interaktionsbildes wird daher eine Trennung zwischen dem Szenengraph, für die visuelle Struktur des Objektraums, und dem MosaIK-Graph, für den strukturellen Aufbau der Bildräume unter Einbeziehung der computergrafischen Kamera, vorgenommen. Vergleichbar mit der strukturellen Organisation eines Datenbildes vollzieht sich der Aufbau des Dialogbildes durch eine baumartige Struktur bestehend aus MosaIK-Knoten, wodurch eine inhärente Strukturierung der Interfaceelemente verwirklicht wird. Die Vorteile dieser Trennung von Daten- und Dialogbild liegen in der besseren Wiederverwendbarkeit von Bestandteilen sowie der vereinfachten Wartbarkeit und Austauschbarkeit durch die Separierung. Eine vergleichbare Unterscheidung von Graphen wird bereits in 3D-Beschreibungssprachen, beispielsweise X3D, durch die Trennung in Szenen- und Verhaltensgraph vollzogen und wird zum Beispiel von FIGUEROA für die Definition von szenengraphunabhängigen Interaktionstechniken eingesetzt (vgl. [FIGUEROA U. A. 2002, S. 54 ff.]).

### 5.6.4 Struktur und Veränderlichkeit des MosaIK-Graphs

Der Aufbau der Dialogbildbeschreibung – der MosaIK-Graph – ist in Abbildung 41 schematisch veranschaulicht und wird in Listing 1 durch eine strukturelle Einordnung der Interfaceelemente des MosaIK-Graphs in einer vereinfachten XML-Struktur aufgezeigt. Den Ausgangspunkt bildet der *Dialogbildknoten* als Wurzel des MosaIK-Graphs. Unterhalb dieses Wurzelknotens werden *Rasterknoten* angeordnet, welche die einzelnen Ebenenraster für das Interfaceraster bilden (siehe Unterabschnitt 5.5.6). Auf einem Raster können wiederum *Panel* platziert werden, die als MosaIK-Knoten unterhalb eines Rasterknotens gruppiert sind. In die Panels werden *Bildräume* eingebettet, die als deren Kindknoten in den MosaIK-Graph eingebunden sind. Die *Kameraknoten* abstrahieren die computergrafischen Kameras und bilden die Blattknoten des MosaIK-Graph. Die Kameras ordnen sich dabei dem jeweiligen Bildraum unter. Die Kopplung von Dialogbildknoten sowie die Verknüpfungen dieser

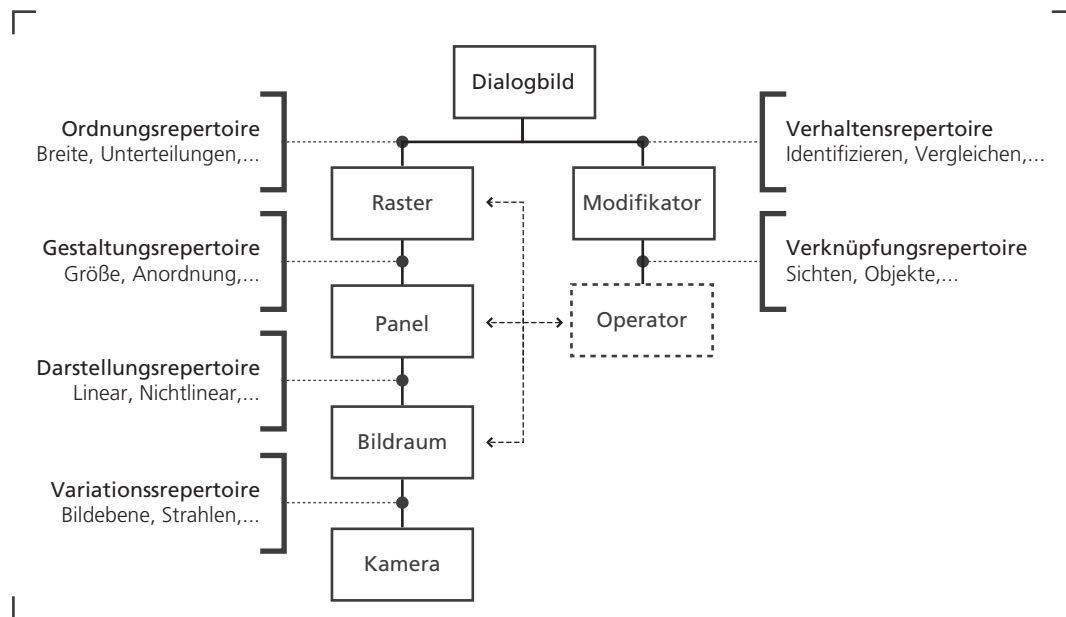


Abbildung 41: Struktureller Aufbau des MosalK-Graphs und das inhärente Gestaltungspotenzial, das für die Beschreibung des Dialogbildes eingesetzt werden kann

```

1| <dialogbild>
2|   <raster DEF= "Raster">
3|     <panel DEF= "Panel">
4|       <bildraum DEF= "bildraum">
5|         <kamera DEF= "Kamera">
6|       </bildraum>
7|     </panel>
8|   </raster>
9|   <modifikator DEF= "Koordination">
10|     <operator DEF= "Kopplung">
11|   </modifikator>
12| </dialogbild>

```

Listing 1: Struktureller Aufbau des MosalK-Graphs als baumartige XML-Struktur

Elemente mit Knoten des Szenengraphs zur Festlegung des Verhaltens einer 3D-Benutzerschnittstelle erfolgen mithilfe von *Modifikatoren*, die sich aus einzelnen *Operatoren* zusammensetzen. Eine Kopplung erfolgt, indem mithilfe der Operatoren Eigenschaftsänderungen oder Abhängigkeiten zwischen den Knoten definiert und zu Modifikatoren gruppiert werden (siehe Unterabschnitt 5.5.8)

Aufgrund der Vielgestaltigkeit des bildgestützten Verhaltens von 3D-Benutzerschnittstellen auf Basis koordinierter multipler Sichten und unter Einbeziehung der Vielfalt an Darstellungsverfahren müssen für eine Beschreibung des Dialogbildes verschiedene Verknüpfungsmöglichkeiten gegeben sein und die Festlegung von Abhängigkeiten definiert werden können. Infolge der angestrebten Trennung von Daten- und Dialogbild umfasst dies einerseits die Verhaltensbeschreibung der Dialogbildelemente untereinander, andererseits die Möglichkeit einer Kopplung von Szenengraph- und Dialoggraph-Elementen. Die nachfolgend vorgestellten Verknüpfungsmöglichkeiten sind zum einen an den ROUTE-Mechanismus sowie an das DEF/USE-Konzept der Szenenbeschreibungssprache X3D angelehnt (vgl. [BRUTZMAN & DALY 2007, S. 71,189]). Das DEF/USE-Konzept beschreibt die Wiederverwendbarkeit von bereits spezifizierten Graphenteilen, während das ROUTE-Konzept von X3D die Veränderbarkeit von



Szenengraphbestandteilen bietet, indem unidirektionale Werteübertragungen zwischen Quell- und Zielknoten umgesetzt werden. Zum anderen werden spezielle Erweiterungen eingeführt, die zur Verbindung von Dialogbildelementen sowie zur Verknüpfung von Knoten des Mosalk-Graphs und des Szenengraphs dienen.

- *Wiederverwendung von Knoten:* Die Funktionalität des Verknüpfungsform basiert auf dem DEF/USE-Konzept von X3D und findet im Mosalk-Graph eine direkte Entsprechung. Dabei wird unter Voraussetzung des gleichen Knotentyps die Werteübertragung eines Quellknotens zu einem Zielknoten vollzogen. Hierbei wird der gesamte Quellknoten – alle Attribute und dessen Werte – auf den Zielknoten abgebildet. Aufgrund dieser Eigenschaft müssen die verknüpften Knoten vom identischen Knotentyp sein, wodurch Verknüpfungen zwischen Knoten des Dialog- und des Szenengraphs ausgeschlossen sind.
- *Wiederverwendung von Attributen:* In einer funktionalen Erweiterung zum beschriebenen USE-Konzept können ferner spezifische Attribute miteinander verknüpft werden. Durch die Wertübertragung von Attributen wird erreicht, dass Quell- und Zielattribut den gleichen Wert besitzen. Für die Verknüpfung von Attributen zwischen Mosalk-Knoten müssen die gekoppelten Datentypen aufeinander abbildbar sein.
- *Verknüpfung von Attributen:* Die Verknüpfung von Attributen basiert auf dem ROUTE-Konzept von X3D. Hierbei handelt es sich um eine ereignisbasierte Verknüpfungsform auf Basis derer eine unidirektionale Werteübertragung zwischen einem Quell- und einem Zielknoten ermöglicht wird. Die Datentypen der Attribute müssen für eine Kopplung übereinstimmen.
- *Verknüpfung von Elementgruppen:* Aufbauend auf dem Prinzip der Attributverknüpfung ist eine Kopplung von Elementgruppen möglich. Diese Verknüpfungsart kann eingesetzt werden, um komplexe Kopplungen zu beschreiben. Beispielsweise hat die Erzeugung von Sichten durch mehrere Kameras zur Folge, dass jede Kamera individuelle Parameter besitzt, jedoch die Einzelkameras als ein gemeinsames Kameramodell behandelt werden müssen. Durch die Verknüpfung werden die Wertänderungen innerhalb einer Elementgruppe relativ oder absolut auf die Elementgruppe des Zielknotens übertragen, wodurch beispielsweise Positions- und Ausrichtungsveränderungen der Kameras synchronisiert werden können. Bei dieser Verknüpfungsform müssen ebenfalls die Datentypen der Attribute identisch sein.
- *Verknüpfung für die Bildsynthese:* Diese Verknüpfung dient der Zusammenführung von Elementen des Szenen- und des Mosalk-Graphs. Die Kopplung ermöglicht, für den Vorgang der Bildsynthese, die Zuordnung eines Szenengraphknoten inklusive dessen Subhierarchien zu einem Mosalk-Knoten. Dies ist beispielsweise für die Zuordnung von Szenengraphelementen zu spezifischen Kameras zur Erzeugung einer objekt-raumbasierten Multiperspektive notwendig. Durch eine Zuweisung einzelner Szenengraphbestandteile zu einem Element des Kameramodells sind diese exklusiv zugeordnet. Die Verknüpfung von zwei Mosalk-Knoten ist hierbei ausgeschlossen.

## 5.6.5 MosalK-Graph-Knotenmenge

Auf Basis des XML-Schemas MosalK-Knoten wird nunmehr eine Knotenmenge für die Beschreibung von Dialogbildern spezifiziert. Die nachfolgend beschriebenen Knoten sind in Abbildung 41 aufgeführt und bilden den Elementfundus zur Bildung von MosalK-Graphen. Für ein besseres Verständnis werden die Knoten in schematischer Form verbildlicht und die beschriebenen Attribute aufgeführt und näher erläutert. Ferner werden wichtige Zusammenhänge der Knoten im Hinblick auf die Strukturierung in einem MosalK-Graphen dargelegt.

### 5.6.5.1 Das Raster

Für die Anordnung mehrerer Sichten auf einer Darstellungsfläche wird das Konzept des Grundlinienrasters, welches als Ordnungssystem für die Organisation mehrerer computergrafischer Darstellungen im 3D-Interface dient, spezifiziert. Mithilfe der Knotendefinition *Raster* wird festgelegt, dass ein Grundlinienraster auf der Darstellungsfläche verortet wird, indem dieses durch eine *Position* (*xPos*, *yPos*) und eine *Größe* (*xSize*, *ySize*) definiert wird. Die Festlegung der *Höhe* (*ySize*) und *Breite* (*xSize*) bestimmt die Dimensionen eines Ebenenrasters, das in dessen größter Ausdehnung durch die Abmessungen der Darstellungsfläche begrenzt wird. Darüber hinaus wird durch die Knotenattribute *Reihe* (*rows*) und *Spalte* (*columns*) die Unterteilung des Rasters vorgegeben. Da im Zuge der Strukturierung einer Benutzerschnittstelle mehrere Ebenenraster in einem Interfacerraster eingesetzt werden können (siehe Unterabschnitt 5.5.6), wird das Attribut *Ebene* (*layer*) in der Knotendefinition statuiert. Durch diese Eigenschaft des Rasterknotens kann eine Sichtbarkeitsreihenfolge festgelegt werden, indem durch das Ebenenattribut eine Staffelung im Interfacerraster vorgegeben wird. In der Folge können optisch überlappende, ineinander geschachtelte oder gestaffelte Ebenenraster erzeugt werden. Für die Gewährleistung einer Flexibilität des spezifizierten Rasterknotens im interaktiven Kontext einer 3D-Anwendung ist dessen initiale Parameterfestlegung nicht unumstößlich. Beispielsweise kann es zur Unterstützung des interaktiven Prozesses eines Nutzers mit der Benutzerschnittstelle notwendig sein, eine geänderte Sichtbarkeitsreihenfolge oder eine angepasste Rasterisierung bereitzustellen. Hierdurch können zur Laufzeit Sichten erzeugt und entfernt werden, um eine Informationsmenge zu adaptieren sowie Hervorhebungen oder Zurückstellungen in Abhängigkeit der Relevanz von Sichten im Nutzer-Bild-Dialog zu verwirklichen. Die Abbildung 42 zeigt zusammenfassend die Spezifikation des Rasterknotens in einer schematischen Darstellung und verdeutlicht dessen grafische Ausgestaltung, während die Tabelle 9 weiterführend die Attributnamen des Knotens und deren Bedeutung tabellarisch aufführt.

### 5.6.5.2 Das Panel

Der MosalK-Knoten *Panel* spezifiziert eine gerahmte Fläche, die auf einem Raster positioniert werden kann. Das Panel fungiert somit einerseits als Grundfläche für ein computergrafisches Bild, andererseits dessen Rahmen als Trennelement zwischen den Bildräumen. Durch das Arrangieren von gerahmten Flächen auf einem Raster ist eine Metaebene zur Strukturierung von Elementen des Dialogbildes in einer 3D-Benutzerschnittstellen gegeben. Die Zuordnung

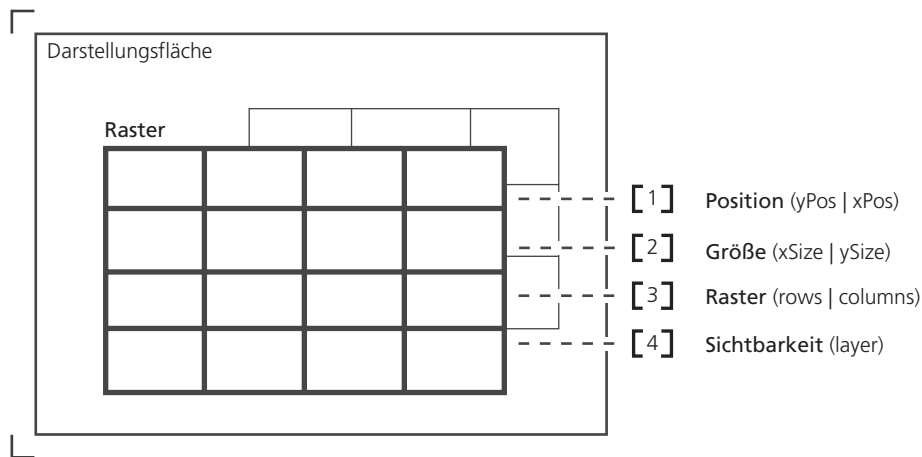


Abbildung 42: Schematische Darstellung des Rasterknotens und dessen Attribute für die Beschreibung eines Grundlinienrasters auf einer Darstellungsfläche

Attribute (Raster)	Beschreibung
xPos	Die horizontale Position auf der Darstellungsfläche in Pixel (linke obere Ecke)
yPos	Die vertikale Position auf der Darstellungsfläche in Pixel (linke obere Ecke)
xSize	Die Breite des Rasters in Pixel
ySize	Die Höhe des Rasters in Pixel
rows	Die Anzahl der Reihen des Rasters
columns	Die Anzahl der Spalten des Rasters
layer	Die Festlegung der Sichtbarkeitsreihenfolge durch Ebenen

Tabelle 9: Übersicht über die Attribute des Rasterknotens des Mosalk:Dialogs

von Panels zu einem Raster wird durch den hierarchisch-strukturierten Aufbau in Form einer Eltern-Kind-Relation im Mosalk-Graph erreicht. Darüber hinaus kann ein Panel auf mehreren Rastern platziert werden, indem durch die Verwendung des *USE*-Elementes ein definiertes Panel als Kindknoten verschiedener Rasterknoten eingeordnet wird. Die Position eines Panels auf dem Raster wird durch die Angabe von *Reihe* (*rowPos*) und *Spalte* (*colPos*), ausgehend von der linken oberen Ecke, auf einem Raster beschrieben. Diese relative Positionierung dient der Flexibilität, da bei einem veränderten Ausgabegerät keine Anpassung der Panelknoten-Werte notwendig wird. Dies erhöht im Vergleich zu einer absoluten Positionsangabe des Panels die Wiederverwendbarkeit der Benutzerschnittstellenbeschreibung. Die Größe der gerahmten Fläche wird ferner durch die Ausdehnung auf dem Raster festgelegt. Mithilfe die Attribute *rasterRowSpan* und *rasterColSpan* wird die Anzahl der Zeilen und Spalten definiert, über die sich das Panel auf dem Raster erstreckt. Infolge dieser Festlegung ist zusätzlich zur Position eines Panels auch dessen Ausdehnung vom Raster abhängig und gleichzeitig von der zugrunde liegenden Darstellungsfläche entkoppelt.

Die Komposition von Panels auf der Darstellungsfläche, unter Berücksichtigung von deren Größe und deren Abstand zueinander, hat einen entscheidenden Einfluss darauf, wie der Nutzer die einzelnen Sichten innerhalb der Benutzerschnittstelle in Zusammenhang bringt. Die Charakterisierung eines Panels als rechteckige Fläche bildet dabei eine Grundlage für eine effiziente Gestaltung auf überwiegend rechteckigen Ausgabegeräten. Ferner hat, in Bezug

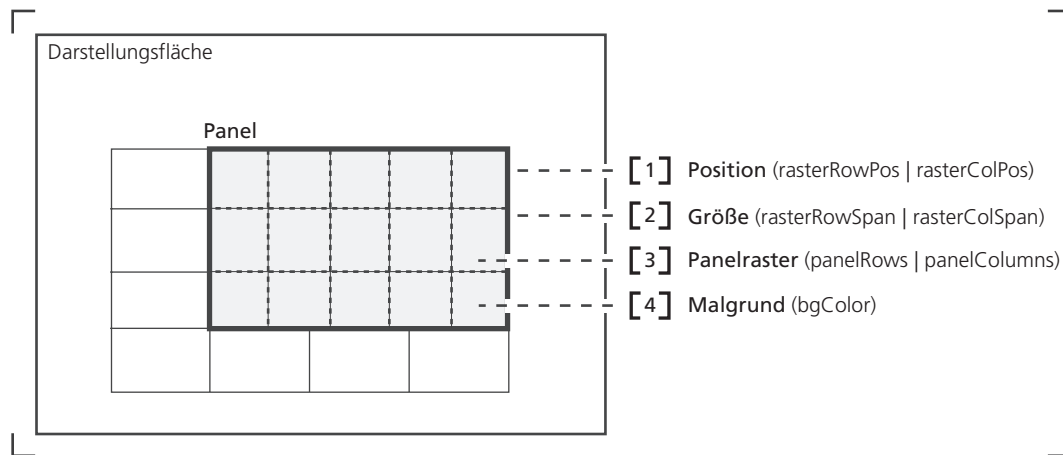


Abbildung 43: Schematische Darstellung des Panelknotens und dessen Attribute für die Beschreibung eines Panels, das auf einem Grundraster platziert wird

Attribute (Panel)	Beschreibung
rasterRowPos	Die horizontale Position des Panels im Raster
rasterColPos	Die vertikale Position des Panels im Raster
rasterRowSpan	Die Anzahl der Spalten über die sich das Panel erstreckt
rasterColSpan	Die Anzahl der Reihen über die sich Panel erstreckt
panelRows	Die Anzahl der Reihen auf der Panelfläche
panelColumns	Die Anzahl der Spalten auf der Panelfläche
bgColor	Farbe der Panelfläche

Tabelle 10: Übersicht über die Attribute des Panelknotens des Mosalk:Dialogs

auf die Panelknotendefinition, der darzubietende Bildraum, insbesondere dessen Größe und Position innerhalb des Panels, einen wesentlichen Einfluss auf den Nutzer-Bild-Dialog. Für die Platzierung eines Bildraums innerhalb des Panels wird die gerahmte Grundfläche in der Paneldefinition als ein „Malgrund“ festgelegt, der mit einer *Farbe* (*bgcolor*) versehen werden kann. Zusätzlich ist, vergleichbar mit einem Ebenenraster, auf dem Malgrund des Panels ein Raster platziert. Die Struktur der Rasterfläche wird durch die Parameter *panelRows* und *panelColumns* beschrieben, wodurch ein Ordnungssystem zur Organisation von Bildräumen geschaffen wird.

Die Darstellung in Abbildung 43 veranschaulicht den Aufbau des Panelknotens, der auf einem Raster positioniert ist und sich über drei Spalten und drei Reihen des Rasters erstreckt. Weiterhin wird in der Abbildung der ebenfalls gerasterte Malgrund dargestellt, der über ein Drei-Mal-Fünf-Panelraster verfügt. Die Attribute des Panelknotens werden zusammenfassend in der Tabelle 10 aufgeführt.

### 5.6.5.3 Der Bildraum

Der *Bildraumknoten* dient der Platzierung eines Bildes innerhalb des Panels. Ein Bildraum wird über die Einordnung als Kindknoten innerhalb des Mosalk-Graphs einem Panel zugeordnet. Durch das *USE*-Konstrukt kann, vergleichbar zum Verhältnis von Panel und Raster, ein Bildraum in verschiedenen Panels dargeboten werden. Für die Charakterisierung des Bildraums

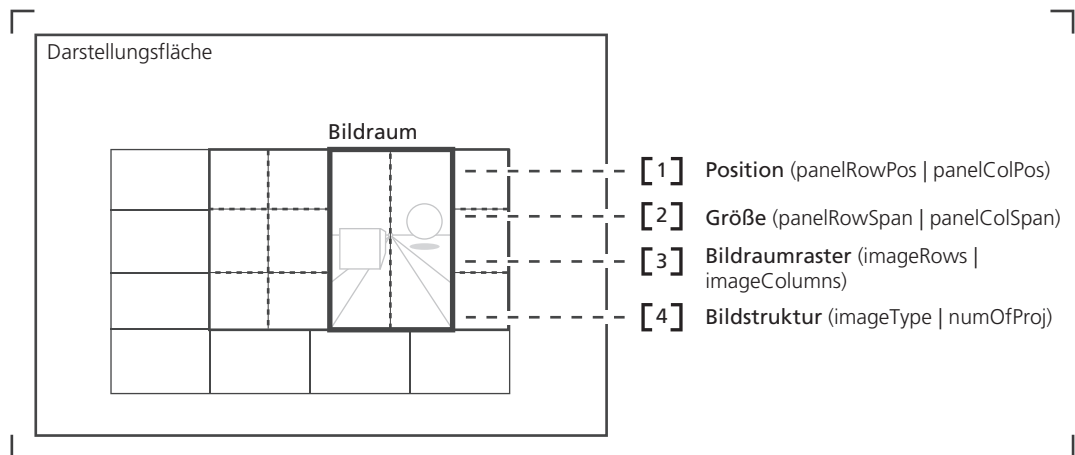


Abbildung 44: Schematische Darstellung des Bildraumknotens und dessen Attribute für die Beschreibung eines Bildraums auf dem Malgrund eines Panels

Attribute (Bildraum)	Beschreibung
panelRowPos	Die horizontale Position des Bildraums im Panel
panelColPos	Die vertikale Position des Bildraums im Panel
panelRowSpan	Die Anzahl der Spalten über die sich der Bildraum erstreckt
panelColSpan	Die Anzahl der Reihen über die sich der Bildraum erstreckt
imageRows	Die Anzahl der Reihen auf dem Bildraumgrund
imageColumns	Die Anzahl der Spalten auf dem Bildraumgrund
imageType	Die Festlegung des Bildtyps <ul style="list-style-type: none"> <li>- Lineare Projektion</li> <li>- Nichtlineare Projektion</li> <li>- Multiprojektion</li> </ul>
projCount	Die Anzahl der zur Erzeugung des Bildraums einzusetzende Kameras

Tabelle 11: Übersicht über die Attribute des Bildraumknotens des MosaikK:Dialogs

wird die entwickelte Systematisierung der Projektionsverfahren zugrunde gelegt (siehe Unterabschnitt 3.3.3). Darauf aufbauend werden in der Knotendefinition Attribute eingeführt, welche die Darstellungsmethode und die Anzahl der zu verwendeten Projektionen festlegen. Dazu ist mithilfe des Attributes *imageType* die Art der computergrafischen Projektion beschreibbar und entsprechend der Klassifikation (siehe Unterabschnitt 3.3.3) durch eine Festlegung auf *lineare (linear)*, *nichtlineare (nonlinear)* oder *multiple (multiple)* Projektionsverfahren definiert. Zusätzlich wird die *Anzahl der Projektionen (projCount)*, durch die der Bildraum erzeugt wird, festgelegt. Ein durch diese Attribute definierter Bildraum wird auf dem Malgrund eines Panels platziert. Dieser muss sich nicht notwendigerweise über die gesamte Größe des Panels erstrecken. Durch das Raster, welches sich auf dem Malgrund des Panels befindet (siehe Abbildung 43), kann mithilfe der Attribute *panelRowPos* und *panelColPos* die Platzierung des Bildraums innerhalb des Panels und durch die Attribute *panelRowSpan* und *panelColSpan* die Ausdehnung des Bildraums festgelegt werden. In der Folge lässt sich ein 3D-Interface nicht nur mit den Bildräumen, sondern ebenso mit den Bildzwischenräumen gestalten. Für eine Organisation von Einzelbildern in einer multiprojektiven Darstellung wird, vergleichbar mit den Attributen des Raster- und Panelknotens, der Bildraum

mit einem Ordnungsraster versehen. Hierbei dienen die Attribute *imageRows* und *imageColumns* zur Festlegung der *Zeilen* und *Spalten* des Bildraumrasters.

Die Attribute des Bildraumknotens werden in Abbildung 44 schematisch und in Tabelle 11 durch die Auflistung der Attribute sowie einer zugehörigen Beschreibung zusammengefasst.

#### 5.6.5.4 Die Kamera

Die Beschreibung eines computergrafischen Bildraums erfolgt durch die Festlegung eines Kameraabbildes beziehungsweise durch die Komposition mehrerer Kameraabbilder zu einem Bild. Die Parameter der computergrafischen Kamera, die zur Charakterisierung von linearen, nichtlinearen und multiprojektiven Darstellungsweisen notwendig sind, finden nunmehr als Attribute, innerhalb des *Kameraknotens*, Eingang in die Dialogbildbeschreibung. Die Einbindung einer Kamera zur Erzeugung eines Bildraumes erfolgt durch deren Einordnung als Kindknoten unterhalb eines Bildraumknotens, wodurch eine konkrete Zuordnung erreicht wird. Darüber hinaus kann die Definition eines Kameraknotens für mehrere Bildräume erfolgen, indem mithilfe des *USE*-Elementes eine Kameradefinition für die Festlegung weiterer Bildräume übernommen wird.

Der Kameraknoten umfasst zum einen die grundlegenden Kameraeigenschaften für eine zentralprojektive Abbildung, die bereits in bestehenden Beschreibungssprachen, beispielsweise X3D und COLLADA, spezifiziert sind. Diese Parameter der Kamera werden im Element *Kameravektor* (*vector*) zusammengefasst. Durch die Attribute des Kameravektors werden die *Position* (*position*), der *Sichtvektor* (*direction*) und der *Aufwärtsvektor* (*up*) einer Kamera sowie der *Kameraöffnungswinkel* (*horizontalFOV*) definiert, infolgedessen grundlegende Eigenschaften zur Konfiguration einer computergrafischen Kamera in der Knotendefinition verankert sind. Zum anderen wird in der Knotendefinition das Element für den Darstellungsbereich spezifiziert. Durch das *Viewport*-Element mit dessen Attributen *imageRowPos* und *imageColPos* wird die Position des Abbildes eines computergrafischen Projektionsschrittes auf dem Bildraumraster bestimmt. Durch die Attribute *imageRowSpan* und *imageColSpan* wird ferner dessen Ausdehnung festgelegt. Dadurch ist die Einbindung mehrerer Abbilder auf dem Raster eines Bildraums als bildraumbasierte multiperspektivische Darstellungen beschreibbar. Damit das Projektionsverfahren von einer Kamera festgelegt werden kann, wird mit dem Element *projection*, respektive dessen Attribut *cameraType* eine Auswahl von Projektionsmethoden vordefiniert. Dadurch kann das zu erzeugende Bild im Hinblick auf dessen bildstrukturelle Eigenschaften charakterisiert werden. Innerhalb des *cameraType*-Attributes wird zwischen *perspektivisch* (*linear*), *orthografisch* (*orthographic*), *invers-perspektivisch* (*inverseLinear*), *zylindrisch* (*cylindrical*) und *sphärisch* (*spherical*) unterschieden. Im Hinblick auf die Eigenschaften der bereitgestellten Projektionsverfahren wird der Kameraknoten um projektionsspezifische Attribute erweitert. Einerseits wird für die Panoramaprojektion zusätzlich zum horizontalen Kameraöffnungswinkel der *vertikale Öffnungswinkel* (*verticalFOV*) als Attribut hinzugefügt. Andererseits ist für die linearen Darstellungsverfahren eine Positionsbestimmung der *geometrischen Mitte* (*principalPoint*) in der Knotendefinition berücksichtigt. Die Positionierung des Hauptpunktes in linearen Abbildungsverfahren wird mithilfe der

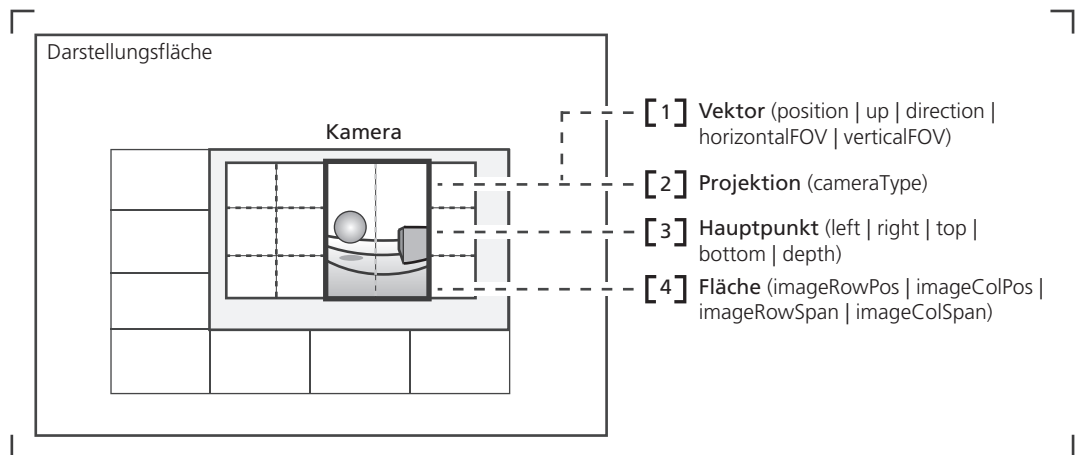


Abbildung 45: Schematische Darstellung des Kameraknotens und dessen Attribute für die Beschreibung eines Abbildes in einem Panel

Attribute (Kamera)	Beschreibung
vector	
position	Die Position der Kamera im Datenraum
horizontalFOV	Der horizontale Kameraöffnungswinkel
verticalFOV	Der vertikale Kameraöffnungswinkel
up	Der Aufwärtsvektor der Kamera
direction	Der Sichtvektor der Kamera
projection	
cameraType	Die Abbildungsmethode im Projektionsschritt der Bildsynthese <ul style="list-style-type: none"> <li>- Zentralprojektion (linear)</li> <li>- Parallelprojektion (orthographic)</li> <li>- Inverse Zentralprojektion (inverseLinear)</li> <li>- Panoramaprojektion (cylindrical)</li> <li>- Fischaugenprojektion (spherical)</li> </ul>
viewport	
imageRowPos	Die horizontale Position des Abbildes auf dem Bildraumraster
imageColPos	Die vertikale Position des Abbildes auf dem Bildraumraster
imageRowSpan	Die Anzahl der Spalten über die sich das Abbild auf dem Bildraumraster erstreckt
imageColSpan	Die Anzahl der Reihen über die sich das Abbild auf dem Bildraumraster erstreckt
principlePoint	
left	Der Abstand des Hauptpunktes vom linken Rand
right	Der Abstand des Hauptpunktes vom rechten Rand
top	Der Abstand des Hauptpunktes vom oberen Rand
bottom	Der Abstand des Hauptpunktes vom unteren Rand
depth	Die relative Position des Hauptpunktes in der Tiefe des Objektraums

Tabelle 12: Übersicht über die Attribute des Kameraknotens des MosaIK:Dialogs



Attribute *left*, *right*, *top* und *bottom* ausgedrückt. Ferner wird mit dem Attribut *depth* die Verortung des Hauptpunktes in der Tiefe definiert. Dies ermöglicht die Erzeugung von zentralprojektiven sowie invers-zentralprojektiven Darstellungen in Abhängigkeit von der definierten Tiefenposition des Projektionszentrums der Kamera.

Die Tabelle 12 fasst die Eigenschaften des Kameraknotens zur Beschreibung von linearen und kurvilinearen Projektionen mit einer tabellarischen Aufzählung der Elemente und Attribute sowie deren Beschreibung zusammen. Die Aufteilung der Kameraeigenschaften in die beschriebenen Elementgruppen ergibt sich einerseits aus der Zusammenfassung semantisch verknüpfter Parameter einer computergrafischen Kamera, andererseits aus der Bereitstellung einer flexiblen Verknüpfbarkeit von Eigenschaften im Hinblick auf die Dynamisierung und Kopplung von Sichten. Die Abbildung 45 bietet ferner eine schematische Darstellung der Attributgruppen des Kameraknotens.

### 5.6.5.5 Der Modifikator

Das Verhalten einer 3D-Benutzerschnittstelle, das aus den Parameterveränderungen der Raster- Panel-, Bildraum- und Kameraknoten resultiert, wird durch den *Modifikatorknoten* festgelegt. Dieser kann das Verhalten eines einzelnen Knotens definieren und Verknüpfungen zwischen Knoten oder Attributen festlegen. Die Spezifikation des Knotens bietet die Möglichkeit einer Verhaltensbeschreibung, die das Prinzip des Entwurfsmusters Modifikation (siehe Unterabschnitt 5.5.8) in den Mosalk-Graph überführt. Der Modifikatorknoten ermöglicht daher die Festlegung von bildstrukturellen Veränderungen sowie von Abwandlungen des Rasters und des Panels, indem Wertmodulationen von Knotenattributen durch die Definition von einzelnen Operatoren beschrieben werden. Bei der Definition eines Operators wird unterschieden, ob dieser durch einen festgelegten Eingabeparameter einen Knoten modifiziert oder ob der Operator die Parameter eines Knotens relativ oder absolut zu den Attributwerten eines anderen Knotens verändert. Da bei komplexen interaktiven 3D-Benutzerschnittstellen die Zahl der notwendigen Verknüpfungsanweisungen schnell steigt und folglich zu einer Unübersichtlichkeit im Instanzdokument führen kann, ergibt sich der strukturelle Aufbau eines Modifikatorknotens durch eine Gruppierung von zusammengehörigen Operatoren. Hierdurch wird zusätzlich zur syntaktischen Festlegung, eine Ordnung semantisch zusammengehöriger Verbindungen in einer komplexen Verhaltensbeschreibung erreicht.

Der Modifikatorknoten verfügt einerseits über Attribute, die Knoten, welche in den Veränderungsprozess involvierten sind, in Verbindung setzt. Andererseits besitzt ein Modifikator Attribute, welche die Art und Weise der Veränderung festlegen. Durch das Attribut *type* wird die Art der Verknüpfung festgelegt. Hierbei liegen die Verknüpfungsarten zugrunde, die in Unterabschnitt 5.6.4 erläutert sind. Das Attribut *condition* dient als Prüfparameter und gibt durch dessen Wert Auskunft darüber, ob gegebenenfalls bestehende Bedingungen für die Anwendung eines Modifikators erfüllt sind. Dies ermöglicht die Ausführung eines Verhaltens unter Einhaltung einer festgelegten Vorgabe, wodurch eine Umsetzbarkeit des Entwurfsmusters Wandler (siehe Unterabschnitt 5.5.7) auf Basis der Verknüpfung von Attributen und

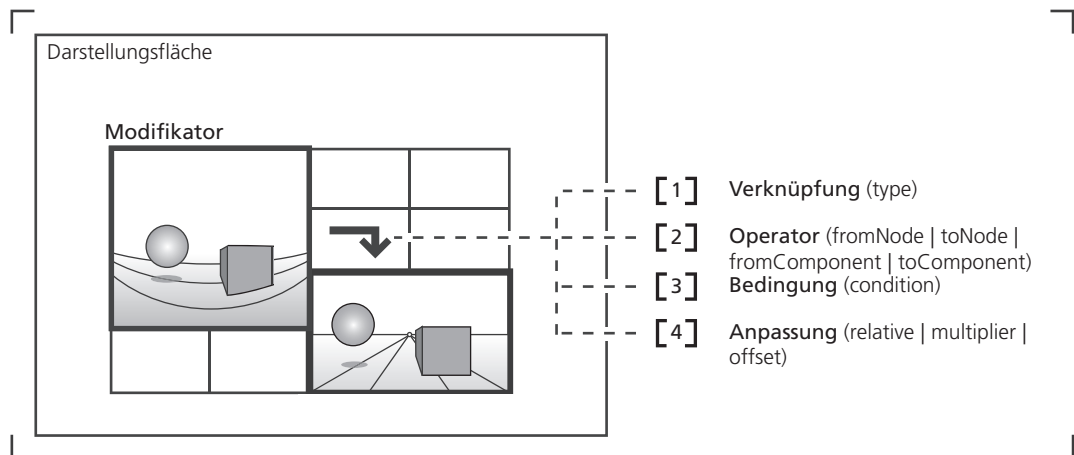


Abbildung 46: Schematische Darstellung des Modifikatorknotens und dessen Attribute für die Beschreibung des Verhaltens von Sichten

Attribute (Modifikator)	Beschreibung
type	Die Festlegung des Verknüpfungstyps
condition	Prüfparameter für Vorbedingungen
operator	
fromNode	Der Senderknoten einer Modifikation
toNode	Der Empfängerknoten einer Modifikation
fromComponent	Das Senderattribut oder das Senderelement eines Knotens
toComponent	Das Empfängerattribut oder das Empfängerelement eines Knotens
relative	Relative oder absolute Veränderung des Attributes oder des Elementes
multiplier	Multiplikator auf eine Operation
offset	Offset auf eine Operation

Tabelle 13: Übersicht über die Attribute des Modifikatorknotens des MosaIK:Dialogs

```

1| <modifier DEF="CoordinatedViews" type="AttributeLink">
2|   <operator fromNode="C1" fromComponent="Position" toNode="C2" toComponent="Position" ...=" " />
3|   <operator fromNode="C1" fromComponent="LookAt" toNode="C2" toComponent="LookAt" ...=" " />
4|   <operator fromNode="C1" fromComponent="UP" toNode="C2" toComponent="UP" ...=" " />
5|   ...
6| </modifier>

```

Listing 2: Definition eines Modifikatorknotens bestehend aus drei Operatoren zur Kopplung von zwei Kameras zur Erzeugung koordinierten multipler Sichten

Elementgruppen sichergestellt werden kann. Die Definition sowie die Überprüfung einer Bedingung sollte in der Umsetzung einer interaktiven Anwendung oder durch die Spezifikation eines Verhaltensgraphs innerhalb einer Szenenbeschreibungssprache erfolgen und ist aufgrund dessen nicht Bestandteil der Knotendefinition. Für die Festlegung des konkreten Verhaltens dient das Element *operator*, welches sich wiederum aus einzelnen Attributen zusammensetzt. Innerhalb des Operators können der Sender (*fromNode*, *fromComponent*) und der Empfänger (*toNode*, *toComponent*) einer Wertänderung definiert werden. Darüber hinaus ist eine Anpassung des Verhaltens durch zusätzliche Parameter in Form eines Versatzes (*offset*) und eines Multiplikators (*multiplier*) gegeben, um eine flexible Verhaltensbeschreibung zu gewährleisten. Durch das Attribut *relativ* kann ferner festgelegt werden,

inwieweit die definierte Wertänderung zwischen Sender und Empfänger relativ oder absolut vollzogen werden soll. Das Listing 2 zeigt, in gekürzter Form, den Quelltext eines Modifikatorknotens, in welcher eine Verknüpfung von Attributen zur Kopplung von zwei Kameras eingesetzt wird. In diesem Beispiel werden die Position und die Ausrichtung von einer Kamera (C1) auf eine zweite Kamera (C2) übertragen.

Der Modifikator bieten die Möglichkeit das Verhalten eines Dialogbildes festzulegen und dient dadurch der Beschreibung einer bildgestützten Interaktion in 3D-Anwendungen, unter Berücksichtigung der Elemente des Dialogbildes und der Bestandteile des Datenbildes. Die Abbildung 46 zeigt hierzu die schematische Darstellung der Funktionsweise eines Modifikators mit dessen Attributen zur Kopplung von Sichten, die auf einem gemeinsamen Raster angeordneten sind. Die Tabelle 13 fasst weiterhin die Attribute des Modifikatorknotens zusammen. Die erweiterten Techniken für die Interaktion des Nutzers mit einer interaktiven 3D-Anwendung sowie die Integration von Interaktionswerkzeugen müssen jedoch in der Ausgestaltung einer interaktiven Anwendung oder durch die Spezifikation eines Verhaltensgraphs erfolgen.

### 5.6.6 Das finale Benutzerschnittstellendesign

Die Beschreibung einer 3D-Benutzerschnittstelle auf Basis des MosalK:Graph enthält keine Informationen in Bezug auf die visuelle Ausgestaltung des Dialogbildes auf dem Ausgabegerät. Daher ist es notwendig, die Beschreibung der 3D-Benutzerschnittstelle in eine konkrete Form zu transformieren. Die Ausgestaltung eines 3D-Interface unter Verwendung der MosalK:Dialog Beschreibungssprache liegt dabei in der Hand von den Entwicklern interaktiver Systeme. Generell bieten sich zwei Möglichkeiten für die visuelle Präsentation des Dokumenteninhaltes als Benutzerschnittstelle einer interaktiven 3D-Anwendung an. Einerseits kann eine Applikation bereitgestellt werden, die auf ein XML-Dokument und dessen Präsentationssemantik spezialisiert ist, oder es erfolgt eine Transformation in ein anderes XML-Format, für das wiederum eine entsprechende Anwendung existiert. Für den zweiten Ansatz wurde die Extensible Stylesheet Language (XSL) [XSL] vom W3-Konsortium [W3C] entwickelt. Auf Basis von Sprachen, die XSL als Metagrammatik nutzen, bieten sich Möglichkeiten bestehende MosalK-Graph-Instanzdokumente in verschiedenartige Dokumente zu konvertieren. Somit ist ein vielfältiger Einsatz in Autorenwerkzeuge oder Anwendungssystemen mit unterschiedlichen Dateiformaten realisierbar. Derzeitige Transformationssprachen bieten vorrangig eine Konvertierung von Geometrieinformationen, während die Bildrauminformationen kaum eine Berücksichtigung finden. Aufgrund dessen wäre die aufwendige Erstellung von Transformatoren, welche ein MosalK-Graph-basiertes Dialogbild als Quelldokument in ein Zieldokument überführen, notwendig, wobei gleichzeitig eine Erweiterung der Beschreibungsmöglichkeiten beziehungsweise eine Spezifikation in der Zielsprache nicht ausgeschlossen werden kann. Für eine Integration und gleichzeitige Überprüfbarkeit der Beschreibungssprache in einer interaktiven 3D-Anwendung ist jedoch die Einbindung in das Bildsprache LiveLab ein zunächst konsequenter und zweckmäßiger Schritt. Die direkte Überführung der Inhalte eines Instanzdokumentes in eine Menge von Benutzerschnittstellenelementen lässt sich in BiLL mithilfe eines Interpreters durchführen, indem eine Komponente anhand der XML-Beschreibung die Elemente der Benutzerschnittstelle instanziiert. Für diese Form der Transformation

nimmt das Bildsprache LiveLab eine entsprechende Interpreterposition ein, wodurch aus der XML-Beschreibung eine konkrete Benutzerschnittstelle in der Arbeitsumgebung erzeugt werden kann. In der Folge bildet das Bildsprache LiveLab ein Werkzeug für die visuelle Vergewärtigung einer 3D-Benutzerschnittstellen auf Basis von Instanzdokumenten, basierend auf dem Mosalk-Vorgehen. Hierzu bietet das POCO-Framework, als ein essenzieller Teil des Bildsprache LiveLab (siehe Unterabschnitt 4.3), die Schnittstellen für das Lesen und Schreiben von XML-Dokumenten. Ferner bietet die bereitgestellte Szenengraph-API ein XML-basiertes Graphenformat für die Einbindung eines Datenbildes sowie Schnittstellen zur Interpretation der Mosalk-Graph-Instanzdokumente zur Erzeugung des Dialogbildes. Die Verwirklichung eines Interaktionsbildes für 3D-Benutzerschnittstellen erfolgt somit durch das Einlesen der Instanzdokumente und die Transformation in eine grafische Benutzerschnittstelle innerhalb des Bildsprache LiveLab.

## 5.7 Resultate und Zwischenfazit

Nach einführenden Erläuterungen in Bezug auf die Herausforderungen im Entwurfsprozess von 3D-Benutzerschnittstellen folgte eine Betrachtung von gebrauchstauglichen Beschreibungsformen für die Unterstützung des Entwicklungsprozesses. Hierbei wurden system- und benutzerzentrierte Ansätze berücksichtigt. Die Reflexion bestehender Verfahrensweisen und Vorgehensmodelle führte zur Konzeption der modellbasierten Vorgehensweise Mosalk. Das Mosalk-Vorgehen unterstützt in einer ersten Phase die Identifikation von Nutzungsanforderungen. Darauf aufbauend kann durch die Modellierung, basierend auf den spezifizierten UML-Profilen Mosalk:Abstrakt und Mosalk:Konkret, die Übersetzung von Anforderungen in eine Modellbeschreibung der 3D-Benutzerschnittstelle erfolgen. Ferner wurden Entwurfsmuster als probates Mittel der Etablierung, Anwendung und Wiederverwendung von Entwurfswissen konzipiert und erläutert, um auf diese Weise die bereits im dritten Kapitel der Arbeit formulierten allgemeinen Gestaltungshinweise hinsichtlich der Ausgestaltung von 3D-Benutzerschnittstellen zu konkretisieren. Für eine zielgerichtete Unterstützung sind die Entwurfsmuster an die Modellierungsphase des Vorgehens angelagert und darüber hinaus auf Basis der UML-Profile beschrieben. In der dritten Phase der Mosalk-Vorgehensweise wurde mit Mosalk:Dialog eine XML-basierte Auszeichnungssprache eingeführt, mit der eine Beschreibung des modellierten 3D-Benutzerschnittstellenentwurfes als graphbasiertes XML-Dokument realisiert werden kann. Die Beschreibung der Benutzerschnittstelle als strukturierter Graph ermöglicht einerseits die notwendige Trennung der Benutzerschnittstelle von der Anwendungslogik, andererseits die Trennung des Dialogbildes vom Datenbild. Der spezifizierte Mosalk-Graph bietet eine syntaktische Grundlage und gleichzeitig eine definierte Knotenmenge für eine Beschreibung des Dialogbildes interaktiver 3D-Interfaces. Dadurch konnte der Umfang der Gestaltungsmöglichkeiten bisheriger Szenenbeschreibungssprachen erweitert und infolgedessen ein erster notwendiger Schritt zu einer umfassenden Beschreibung von Interaktionsbildern für 3D-Anwendungen vollzogen werden.

Nachfolgend werden wesentliche Charakteristiken der modellbasierten 3D-Interfacekomposition (Mosalk) zusammengefasst:

- *Die Notation:* Die Beschreibung von 3D-Interfaces und Entwurfsmustern erfolgt in einer einheitlichen UML-Notation. Darüber hinaus erfolgt die Beschreibung der Benutzerschnittstelle auf Basis des anerkannten XML-Standards.
- *Die Systembildung:* Mithilfe des modellgestützten Vorgehens lassen sich die Phasen des Interfaceentwurfs (von konzeptionell zu final) semantisch verknüpfen und methodisch ineinander überführen.
- *Die Wiederverwendung:* Erfolgreiche Lösungen beim 3D-Benutzerschnittstellenentwurf können wiederverwendet werden, wodurch Quasi-Standards etabliert werden können.
- *Die Qualitätssicherung:* Die Einführung von Musterlösungen trägt dazu bei, dass etablierte Konzepte des 3D-Interfacedesigns als wiederverwendbare Vorlagen zur Erfüllung softwareergonomischer Anforderungen interaktiver Anwendungen eingesetzt werden können.
- *Die Kommunikation:* Der Entwurf von 3D-Benutzerschnittstellen, der zunächst informellen, darauf aufbauend modellhaften und abschließend deklarativen Charakter besitzt, bildet die Basis für einen interdisziplinären Kommunikationsprozess der beteiligten Personengruppen. Durch diesen kann bestehenden Verständigungsschwierigkeiten entgegengewirkt werden.
- *Die Entwicklungsrelevanz:* Der modellgestützte Entwurf, ebenso wie die Entwurfsmuster, beinhalten auf konkreter Ebene softwaretechnische Implikationen (Empfehlungen hinsichtlich entsprechender Softwarekomponenten). Dadurch wird eine Verbindung zwischen benutzerfreundlichem Design und der strukturierten Softwareentwicklung erreicht.
- *Die Werkzeugintegration:* Durch die UML-basierte Beschreibung einerseits und die XML-basierte Spezifikation andererseits ist eine Integration in bereits bestehende Entwurfs- und Entwicklungswerkzeuge möglich.

## 6 MosalK und BiLL in der Anwendung

Die Anwendungsdomänen interaktiver 3D-Systeme haben sich in den letzten Jahrzehnten grundlegend verändert und immens erweitert. Waren es in der Anfangszeit computergrafischer Darstellungen noch Spezialisten, die im Rahmen der Forschung rechnergestützte Verfahren zur grafischen Auswertung von Daten einsetzten, erfolgt die Nutzung von 3D-Anwendungen in der heutigen Zeit in fast allen Bereichen der Wirtschaft, Wissenschaft und Unterhaltung. In der Folge ist es erforderlich, die verschiedenen Personenkreise mit ihren unterschiedlichen Vorkenntnissen und ihrem inhomogenen Verständnis für Algorithmen und Computertechnik in die Lage zu versetzen, interaktive 3D-Anwendungen effektiv und effizient zu nutzen. Durch die aufgezeigten Manipulationsmöglichkeiten der computergrafischen Kamera (Kapitel 3) und deren softwaretechnische (Kapitel 4) sowie methodische Einbindung (Kapitel 5) in den 3D-Interfaceentwurf, ist ein Vorgehen aufgezeigt, das Ingenieurern und Gestaltern die notwendigen Techniken, Methoden und Werkzeuge für die Entwicklung von 3D-Benutzerschnittstellen bereitstellt. Darauf aufbauend wird nunmehr ein Einblick in unterschiedliche Anwendungsbereiche gegeben, in denen die aufgezeigten Verfahrensweisen zur Verwirklichung von interaktiven Benutzerschnittstellen genutzt werden können. Der Fokus der Betrachtung liegt dabei auf dem Bereich wissenschaftlicher Visualisierungen. Dieses Anwendungsgebiet für computergrafische 3D-Darstellungen vermag einerseits die Realisierungsmöglichkeiten zur zielgerichteten Umsetzung intendierter Mensch-Maschine-Kommunikation zu illustrieren, andererseits die Eignung von 3D-Benutzerschnittstellen zur Darstellung heterogener Daten in interaktiven Anwendungen zu reflektieren. Dazu werden konventionelle 3D-Benutzerschnittstellen aufgezeigt, welche die derzeitige grafische Unterstützung in unterschiedlichen Forschungszweigen veranschaulichen. Weiterhin werden Entwürfe von 3D-Interfaces unter Einsatz linearer und nichtlinearer Abbildungsverfahren sowie koordinierter multipler Sichten vorgestellt. Die Ausführungen verdeutlichen, dass durch das MosalK-Vorgehen ein zusätzlicher Nutzen für die Gestaltung interaktiver 3D-Systeme im jeweiligen Forschungsbereich erreicht wird. Für die Realisierung der 3D-Interfaces wird das Bildsprache LiveLab unter Einbindung unterschiedlicher Visualisierungsverfahren eingesetzt. Zur Verdeutlichung der Praktikabilität des MosalK-Vorgehens für die Entwicklung von Benutzerschnittstellen wird zunächst das grafische Erscheinungsbild der interaktiven Anwendung Bildsprache LiveLab vorgestellt (6.1). Ferner wird die Anwendbarkeit des modellbasierten Vorgehens exemplarisch anhand eines Interfaceentwurfes für einen Einrichtungsplaner aufgezeigt (6.2) und darüber hinaus der Einsatz von modellbasierten 3D-Interfaces im Kontext von zwei wissenschaftlichen Anwendungsszenarien verdeutlicht (6.3).

## 6.1 Das Bildsprache LiveLab im Überblick

### 6.1.1 Charakteristik der Arbeitsumgebung

Im vierten Kapitel wird das Bildsprache LiveLab als Arbeitsumgebung zur Erstellung und Gestaltung von Benutzerschnittstellen für interaktive 3D-Anwendungen eingeführt und dessen komponentenbasierte Softwarearchitektur erläutert. Es handelt sich bei dieser interaktiven Anwendung um eine prototypische Umsetzung, ohne den Entwicklungsstand eines kommerziellen Produktes zu bieten. Die Konzeption und die technische Umsetzung der Anwendung entstanden vor allem als Ergebnis aus den Arbeiten von EBNER (vgl. [EBNER 2007]), WOJDZIAK (vgl. [WOJDZIAK 2007]), und KAMMER (vgl. [KAMMER 2009]). Zu den Resultaten zählen die Grundelemente der interaktiven Plattform; der *BiLL-Viewer* und der *BiLL-Editor* sowie der Erweiterungsmechanismus für die Integration zusätzlicher Visualisierungsbausteine zur Laufzeit der Anwendung. Die Ergänzung von Gestaltungsmöglichkeiten wurde überwiegend in Form studentischer Arbeiten vollzogen (unter anderem [MÜNCH 2008; SCHMIDT 2009; STARKE 2011; WOJDZIAK 2007]). Der aktuelle Entwicklungsstand kann auf der Projektwebseite [AG-TV] nachverfolgt werden.

Das Bildsprache LiveLab als ein Werkzeug für den 3D-Interfaceentwurf bietet dem Anwender einen experimentellen Zugang zu der Analyse, dem Entwurf und der Anwendung von 3D-Benutzerschnittstellen. Hierfür bietet BiLL Designern, Entwicklern und Anwendern unterschiedliche Schnittstellen zur Gestaltung und Entwicklung interaktiver 3D-Interfaces, wodurch ein interdisziplinäres Arbeiten ermöglicht wird (siehe Unterabschnitt 4.3.4). Die Anwendung ist zur Laufzeit hinsichtlich der Funktionalität und des Erscheinungsbildes erweiterbar, jedoch initial als Zwei-Fenster-Anwendung konfiguriert. Durch die Teilung in zwei visuell entkoppelte Fenster wird es dem Anwender ermöglicht, die 3D-Visualisierung des Objektraums, im *BiLL-Viewer*, unabhängig von der 2D-Benutzeroberfläche, dem *BiLL-Editor*, zu verwenden. Die 2D-Bedienoberfläche ermöglicht die werkzeuggestützte Modifikation des Objektraums sowie den Eingriff in die Bildsynthese zur Veränderung bildstruktureller Eigenschaften. Das Editor-Fenster kann infolge der Entkopplung von der 3D-Visualisierung im Bedarfsfall hinzugezogen werden. Dies ermöglicht die Nutzung des BiLL-Viewers in einem Vollbildmodus bei nur einem Ausgabegerät oder eine Erweiterung der 3D-Benutzerschnittstelle auf mehrere Ausgabegeräte.

### 6.1.2 Der Funktionsumfang

Im Abschnitt 4.1 werden die bestehenden Anforderungen an ein Werkzeug für die Entwicklung von 3D-Interfaces formuliert. In Anlehnung an Kriterien und Hinweise in Bezug auf den Entwurf interaktiver Visualisierungssysteme von SHNEIDERMAN [SHNEIDERMAN 2004, KAP. 1.2], HERCZEC [HERCZEC 2006, KAP. 1] sowie PREIM und DACHSELT [PREIM & DACHSELT 2010, KAP. 6] werden nun die Forderungen benannt, welche an die grafische Benutzerschnittstelle des Bildsprache LiveLab gestellt werden.



### 6.1.2.1 Werkzeuge und Funktionen des Editorfensters

- Der Import von 3D-Geometrien, Animationen und Texturen aus verschiedenen Dateiformaten
- Der Export von 3D-Szenen, des Benutzerschnittstellenaufbaus sowie des aktuellen Dialogbildes in ein XML-basiertes Zielformat
- Eine Ansicht der Elemente des Daten- und des Dialogbildes in einer baumartigen Graphenstruktur mit Editiermöglichkeiten
- Die Darstellung relevanter Objektraum- und Bildraumeigenschaften in Bezug auf die zur Verfügung stehenden Manipulationsmöglichkeiten in einer kompakten Form
- Eine grafische Benutzeroberfläche für die Verwaltung der Visualisierungskomponenten und die Darstellung des Zustandes einer Komponente zur Laufzeit
- Die Bereitstellung der Funktionalität von Komponenten über eine grafische Benutzeroberfläche mit unterschiedlichen (nicht restriktiven) Nutzermodi für ein interdisziplinäres Arbeiten.
- Die einfache Manipulierbarkeit von Visualisierungstechniken durch Editoren, bei gleichzeitiger Kontrolle von Details (beispielsweise die Einstellung exakter Parameter)

### 6.1.2.2 Werkzeuge und Funktionen des Darstellungsfensters

- Die 3D-Darstellung eines Objektraums mit einer permanenten Aktualisierung für einen interaktiven, echtzeitfähigen Nutzer-Bild-Dialog
- Die Integration von Werkzeugen (3D-Widgets) zur Manipulation der visuellen Struktur und zu Modifikation von Parametern des Objekt- und des Bildraums
- Die Verfügbarkeit unterschiedlicher Navigationsmöglichkeiten für eine szenenspezifische Anpassung der Interaktionsmöglichkeiten zum Orientieren und Operieren
- Die Integration von Gittertypen und Achsendarstellungen für ein vereinfachtes Interagieren mit Szenenelementen und zur Unterstützung relationaler Nutzeraufgaben
- Die Erstellung von multiplen Sichten und deren Koordination sowie die individuelle Konfiguration des Abbildungsverfahrens für jede Sicht

## 6.1.3 Bestandteile der Arbeitsumgebung

Die Standardkonfiguration des Bildsprache LiveLab, bestehend aus dem Viewer- und dem Editor-Fenster, wird durch die Softwarekomponente BiLL4Desktop spezifiziert (siehe Unterabschnitt 4.3.2). Basierend auf der Grafik-Engine OpenSceneGraph erfolgt die computergrafische Bildsynthese und die daraus hervorgehende Echtzeit-Darstellung des Objektraums im Viewer-Fenster (siehe Abbildung 47). Die Interaktionsmöglichkeiten umfassen einerseits das Orientieren im Objektraum durch integrierte Navigationstechniken, andererseits das Operieren durch einen direkten interaktiven Prozess mit den Elementen der visuellen Struktur sowie

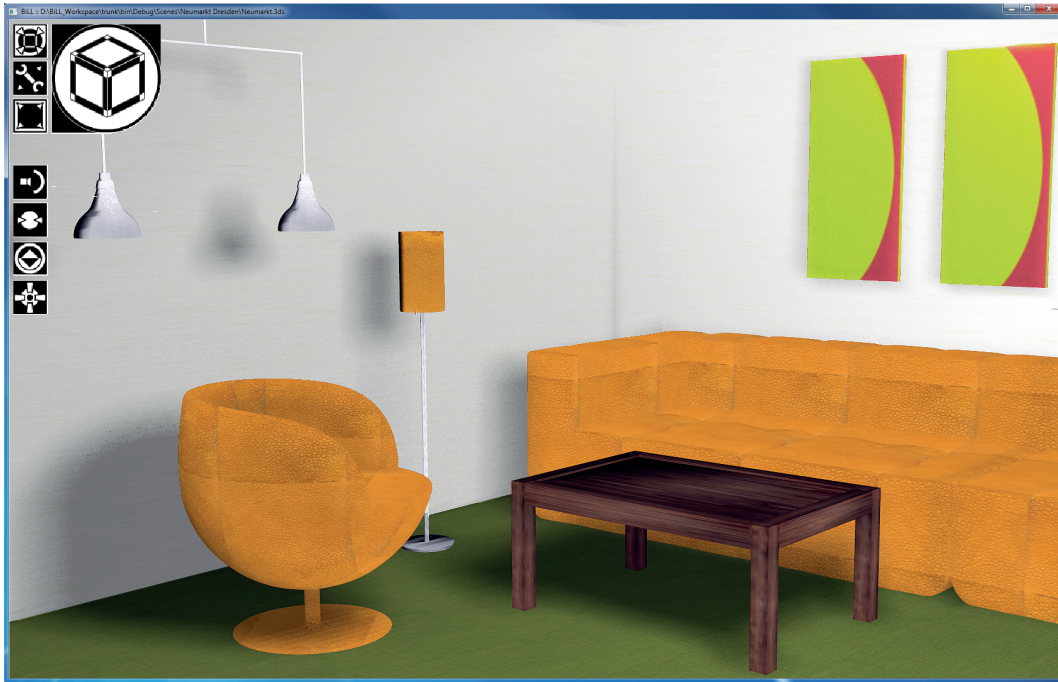


Abbildung 47: Das 3D-Interface des Bildsprache LiveLab (Viewer-Fenster) zeigt einen Objektraum, der einem virtuellen Zimmer eines Einrichtungshauses nachempfunden ist sowie die Elemente des Head-Up-Displays.

einer indirekten Interaktion auf Basis von grafischen Werkzeug- und HUD-Elementen. Das Head-Up-Display ermöglicht die direkte Ausführung von Abbildungsverfahren auf Basis einer vordefinierten Konfiguration der Visualisierungskomponente, wodurch der Bildraum direkt im Viewer-Fenster modifiziert werden kann. Der gebotene Funktionsumfang ist auf die automatische Ausführung des Abbildungsverfahrens beschränkt, um einen Bildraum mit einer vordefinierten Konfiguration zu erzeugen, ohne den Viewer mit einer großen Anzahl von HUD-Elementen zu überlagern. Das Head-Up-Display des Viewers besteht weiterhin aus einem interaktiven Navigationswürfel (siehe Abbildung 47) für die Ausrichtung der Kamera in der Szene. Hierzu ist die Orientierung des Würfels an die kamerabezogene Ausrichtung des Objektraums gekoppelt. Weiterhin sind elementare Funktionen – das Öffnen des Editors-Fensters, das Maximieren sowie das Schließen des Viewer Fensters – in das HUD integriert.

Das zweite Fenster des Bildsprache LiveLab, der Editor, bietet ergänzend eine 2D-Benutzeroberfläche für die Modifikation des Objekt- und des Bildraums (siehe Abbildung 48). Hierzu bietet der Editor grafische Bedienelemente, die innerhalb des Fensters in separaten Bereichen angeordnet sind. Die Oberflächenbestandteile des Editor-Fensters, die als grundlegende Werkzeuge dauerhaft in den Editor integriert sind, umfassen das Menüsystem, die Szenengraphdarstellung, den Komponentenmanager und den Attributeditor (siehe Abbildung 48 oben). Über die editoreigene Menüleiste ist das Laden und Speichern von 3D-Szenen möglich. Darüber hinaus können Einstellungen in Bezug auf das Erscheinungsbild und die Funktionsweise der Arbeitsumgebung vorgenommen werden. Der Szenengraphbereich als ein weiteres Element des Editor-Fensters bietet die geladene 3D-Szene als baumartige Datenstruktur dar. Über die hierarchische Sicht auf die Elemente des Objektraums können

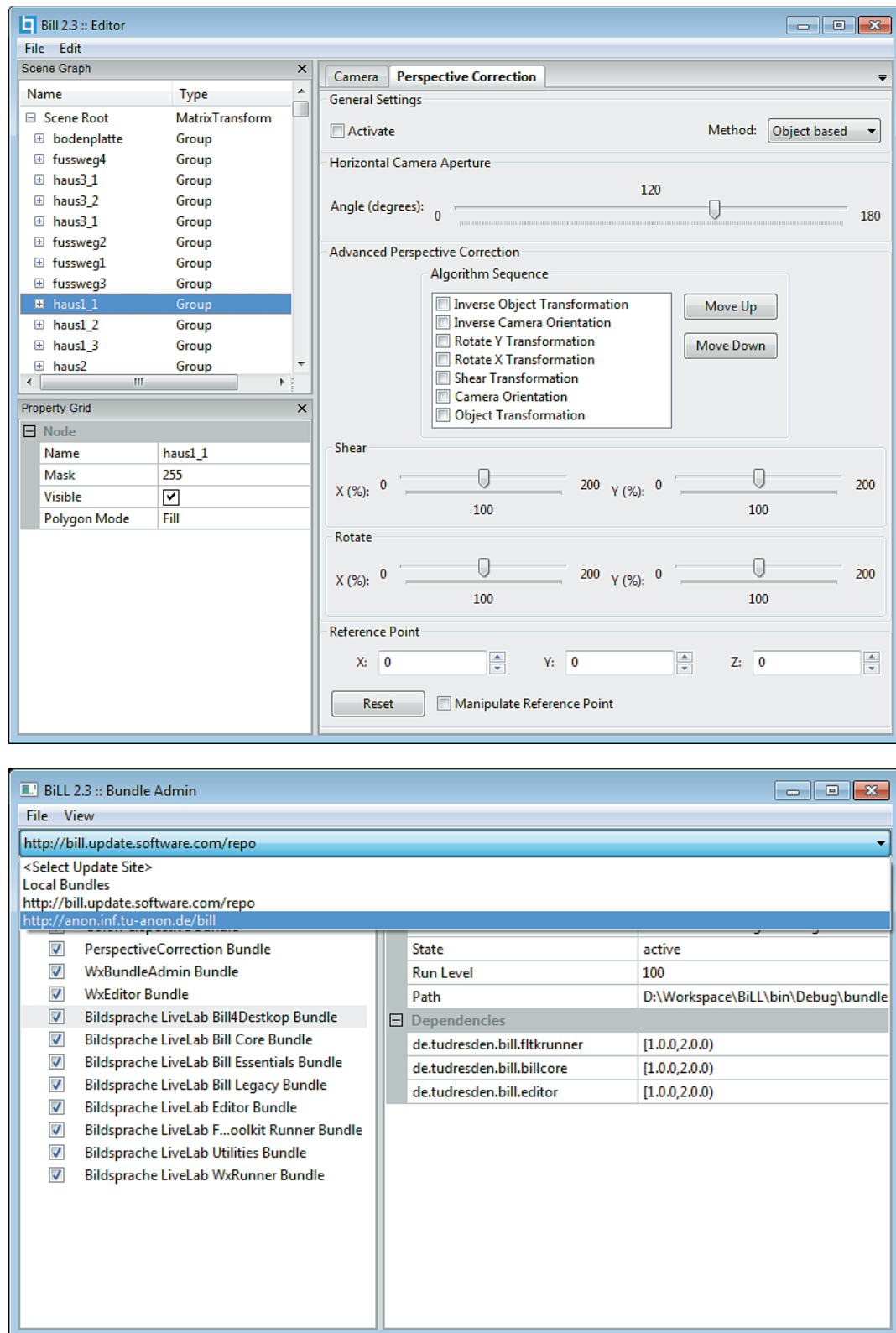


Abbildung 48: Die 2D-Benutzeroberfläche des Bildsprache LiveLab (Editor-Fenster): Die grafische Oberfläche (oben) bietet eine Szenengraphdarstellung, ein Einstellungsfenster und einen Komponentenbereich. Der Bundle Admin (unten) listet bestehende Visualisierungskomponenten auf und bietet Bundle Informationen.

einzelne Objekte für die Bearbeitung selektiert werden. Unterhalb des Szenengraphs ist in der Bedienoberfläche der *Attributeditor* platziert. Dieser ermöglicht den direkten Zugriff auf die Eigenschaften eines selektierten Objektes, wobei sowohl Objekteigenschaften der Primärgeometrie als auch Attribute mit einem Einfluss auf die Sekundärgeometrie editiert werden können. Der rechte Bereich des Editorfensters ist für die Bedienelemente der Visualisierungskomponenten reserviert. Jede Komponente, die über ein grafisches Interface verfügt, wird als eine „Karteikarte“ in die Editor-Oberfläche integriert. Durch die Auswahl eines Karteireiters wird die zugehörige Benutzeroberfläche in den Vordergrund gehoben und steht in der Folge als interaktives Bedienwerkzeug zur Verfügung. Auf diese Weise sind die Visualisierungsbau- steine zur Modifikation der computergrafischen Kamera und der Gestaltung des Bildraums in das Editor-Fenster des Bildsprache LiveLab integriert. Eine genaue Beschreibung der Benutzeroberflächen aller verfügbaren Visualisierungskomponenten und deren Funktionsumfang wird im Benutzerhandbuch des Bildsprache LiveLab dargelegt, das auf der Projektwebseite [AG-TV] zur Verfügung gestellt ist.

Die Verwaltung von Erweiterungskomponenten für die Arbeitsumgebung erfolgt über ein separates Fenster; der sogenannte *Bundle Admin*. Der Administrationsbereich wird über die Menüleiste des Editors aufgerufen und listet alle, zu einem spezifischen Zeitpunkt, inaktiven, installierten, geprüften und aktiven Komponenten des Bildsprache LiveLab auf. Die im Fenster aufgezeigten zusätzlichen Metadaten geben Auskunft über den Entwickler der Komponente, die Version sowie deren Abhängigkeiten zu anderen Komponenten und erleichtern dadurch die Konfiguration des Bildsprache LiveLab. Weiterhin können mithilfe der Komponentenverwaltung die Visualisierungskomponenten hinzugefügt oder entfernt werden, wobei der Lebenszyklus von Bundles nach dem OSGi Standard in der Arbeitsumgebung vollzogen wird (siehe Unterabschnitt 4.2.1).

## 6.2 Fallbeispiel: Produktvisualisierung

Die Präsentation von Produkten in einer virtuellen Umgebung wird bereits in Unterabschnitt 3.1.2 als eine Anwendungsdomäne für interaktive 3D-Anwendungen identifiziert. Die dreidimensionale Darstellung von Gegenständen wird dabei vorwiegend für Werbezwecke und für die Vermarktung von Produkten eingesetzt. In den Präsentationswerkzeugen wird dabei das Produktdesign in den Mittelpunkt gestellt und dessen wesentlicher Einfluss auf die Gebrauchstauglichkeit hervorgehoben. Der Interaktionsgrad ist bei diesen Anwendungen vergleichsweise gering. Im Allgemeinen können Produkte durch den Nutzer von allen Seiten sowie in unterschiedlichen Größen- und Detailstufen betrachtet werden. Zusätzlich können oftmals Beschreibungstexte eingeblendet werden, um dem Kunden das gesamte Produkt oder einzelne Bestandteile in textueller Form näher zu bringen. Weiterhin sind die äußerlichen Eigenschaften, beispielsweise die Farbe oder das Material, eines Produktes konfigurierbar. Mithilfe von steuerbaren 3D-Animationssequenzen können ferner verschiedene Produktfunktionalitäten präsentiert und durch den Nutzer interaktiv getestet werden. Dadurch wird dem Anwender ein Zugang zum Produkt bereitgestellt, welchen dieser sonst nur durch die Interaktion mit dem realen Gegenstand hätte.

Anhand eines Fallbeispiels wird nachfolgend die modellbasierte und mustergestützte Vorgehensweise *MosaIK* für den 3D-Interfaceentwurf erläutert, um dessen praktische Anwendbarkeit zu illustrieren. Dabei dienen Produkte eines fiktiven Einrichtungshauses als Datenbasis für die Präsentationsplattform. In der virtuellen Produktvorstellung, die beispielsweise einen Blick auf das zukünftige Angebot eines Unternehmens bietet, soll es möglich sein, einen eingerichteten Ausstellungsraum zu explorieren, um einen Überblick über die Warenauswahl des Einrichtungshauses zu erhalten. Dem Szenario liegt hierzu ein virtueller Ausstellungsraum – ein Wohnzimmer – zugrunde, in dem Einrichtungsgegenstände platziert sind (siehe Abbildung 47). Das Portfolio des Herstellers bietet zur Ausstattung des Wohnzimmers hierzu Sofas, TV-Möbel, Beistelltische, Lampen und weitere Möbel für die Aufbewahrung. Zusätzlich zur Gesamtansicht des Raumes sollen in diesem Szenario die Möbelstücke und Einrichtungsgegenstände individuell betrachtet werden können. Diese Detailpräsentation soll vor allem dazu dienen, dem potenziellen Kunden die äußere Gestalt und die Ausstattungsmerkmale des Sortimentes zu veranschaulichen.

### Das konzeptionelle 3D-Interfacedesign

Basierend auf dem MosaIK-Vorgehen (siehe Unterabschnitt 5.3) erfolgt in der Spezifikationsphase die Beschreibung von Anwendungsszenarien und die Ableitung von Nutzungsanforderungen an die 3D-Benutzerschnittstelle. Da in der Regel mehrere, auch verbundene, Szenarien ausgearbeitet werden, können diese beispielsweise auf einer virtuellen oder realen Zeichenfläche mithilfe grafischer Elemente arrangiert werden. Die Szenarien können jedoch unabhängig von einer grafischen Unterstützung textuell beschrieben werden, wodurch eine Einschränkung der Nachvollziehbarkeit aufgrund spezieller Beschreibungssprachen bei der Formulierung von Nutzungsanforderungen vermieden wird.

Im gegebenen Fallbeispiel bedingt die Vielzahl an Einrichtungsgegenständen, dass die Darbietung sämtlicher Details jedes Möbelstückes nicht in einer singulären Darstellung erfolgen kann. In der Folge soll die 3D-Benutzerschnittstelle dem Anwender einerseits eine Übersicht über die Produktpalette des Einrichtungshauses bieten, andererseits die Möglichkeit eröffnen, die Möbel im Detail zu betrachten. Daher wird das Konzept einer Übersicht-und-Detail-Darstellung (siehe 3.5.3), wie diese in [CONVERTINO U. A. 2003, S. 56–57] als *Dual View* beschrieben und in [TIDWELL 2011, S. 296 ff.] als Interface-Entwurfsmuster erläutert wird, der interaktiven 3D-Benutzerschnittstelle zugrunde gelegt. Durch diesen Benutzerschnittstellenaufbau wird die Komplexität des Datenbestandes in jeweils fokussierte Bereiche aufgeteilt, wobei deren Zusammenhänge durch die Kopplung der zwei Sichten unterstützt werden soll. Dabei wird die Überblicksdarstellung den gesamten Einrichtungsraum mit den darin befindlichen Möbeln visualisieren, während die Detailansicht die Darstellung einzelner Möbelstücke bietet, die durch den Anwender in der Überblicksdarstellung in den Fokus gesetzt werden.

### Das abstrakte Interfacedesign

Aus den in der Spezifikationsphase identifizierten Nutzungsanforderungen wird in der Modellierungsphase zunächst das abstrakte 3D-Interfacedesign entwickelt. Der Entwurf für die 3D-Benutzerschnittstelle des gegebenen Fallbeispiels ist in Abbildung 49 dargestellt. Dem

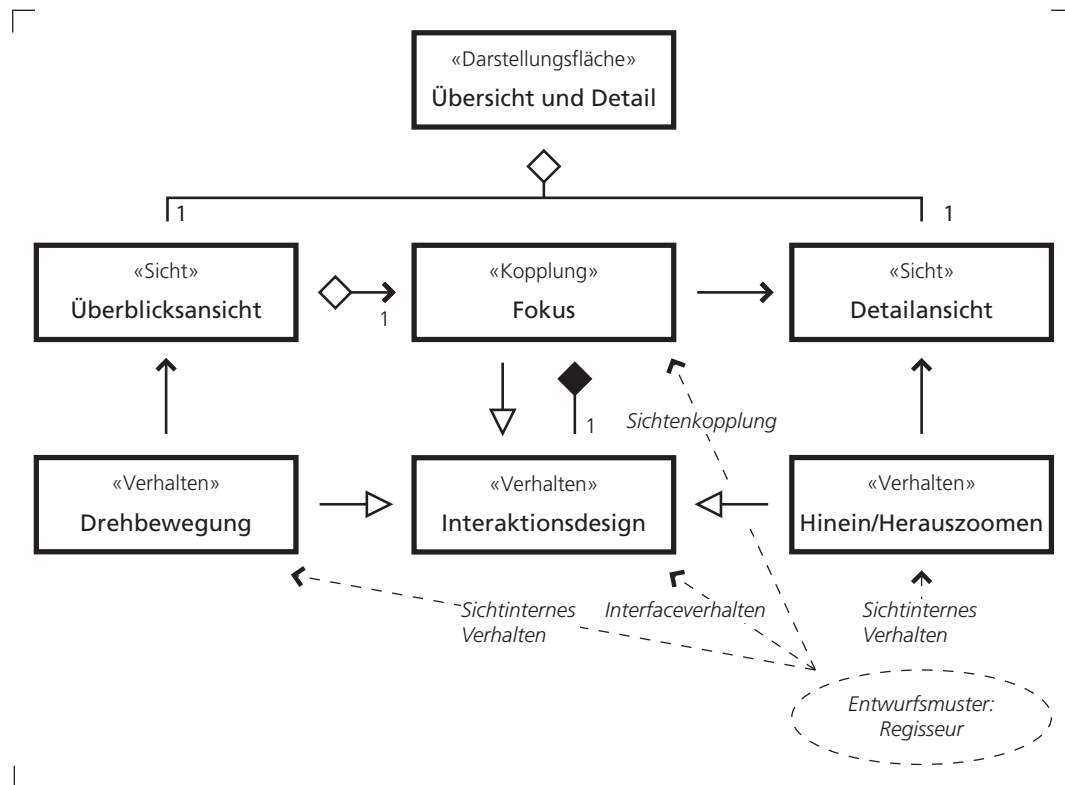


Abbildung 49: Das abstrakte 3D-Interfacedesign für das Übersicht-und-Detail-Szenario auf Basis des MosalK:Abstrakt-Profiles und unter Einbeziehung des Entwurfsmusters Regisseur

Entwurf liegt das MosalK:Abstrakt-Profil (siehe 5.5.1) zugrunde. Die Abbildung zeigt, dass für die Produktpräsentation zwei Sichten – eine *Überblicksansicht* und eine *Detailsansicht* – festgelegt werden, die auf einer Darstellungsfläche (*Übersicht und Detail*) platziert sind. Die Überblicksansicht soll hierbei den Ausstellungsraum zeigen, während in der Detailsansicht ein expliziter Gegenstand visualisiert wird. Für die Festlegung des Sichtverhaltens wird das *Entwurfsmuster Regisseur* als Vorlage verwendet. Basierend auf dem Muster, das die Festlegung des sichtinternen Verhaltens vorgibt und die Vorgehensweise zur Kopplung von Sichten beschreibt, wird die Koordination der zwei Sichten als eine *Master-Slave-Beziehung* (vgl. [ROBERTS 2007, S. 64]) konzipiert. In der Umsetzung des Entwurfsmusters bietet die Überblicksansicht als sichtinternes Verhalten eine *Drehbewegung*, wodurch ein Umsehen im Raum nachgebildet wird. Hierdurch kann eine Sichtanpassung vollzogen werden, bei der Einrichtungsgegenstände nutzerabhängig in die Bildmitte – in den Fokus – gesetzt werden können. In der Detailsansicht ermöglicht das sichtinterne Verhalten eine *Hinein- und Herausbewegung*, wodurch der Anwender den dargebotenen Gegenstand in verschiedenen Größendarstellungen betrachten kann. Durch ein zusätzliches, sichtübergreifendes Verhalten wird jeweils das Objekt in der Detailsansicht visualisiert, welches sich im Fokus der Überblicksansicht befindet.

### Das konkrete Interfacedesign

Im nachfolgenden Entwurfsschritt erfolgt die Ausarbeitung des konkreten 3D-Interfacedesigns. Die Beschreibung der Benutzerschnittstelle für die Produktvisualisierung erfolgt auf Basis des MosalK:Konkret-Profiles (siehe 5.5.4) und wird in Abbildung 50 veranschaulicht. Die



Benutzerschnittstelle, als *Übersicht-und-Detail-Umsetzung*, verfügt über zwei Bildräume, die auf Basis eines *2x1-Rasters* auf einem Interfacebereich verortet sind. Für die Umsetzung der Überblicksdarstellung, mit einer möglichst geringen Verdeckung der dargebotenen Ausstellungsgegenstände, wird anstatt einer konventionellen perspektivischen Sicht eine *Panoramaansicht* eingesetzt, welche die Einrichtungsgegenstände auf Basis einer *Zylinderprojektion* abbildet. In die computergrafische Bildsynthese werden dazu alle *Szenenobjekte* des Ausstellungsraumes einbezogen. Die nutzerabhängige Veränderung des Bildraums wird, entsprechend des abstrakten 3D-Interfaceentwurfs, durch eine *Horizontalrotation* der computergrafischen Kamera verwirklicht.

Damit in der *Detailansicht* möglichst viele Informationen über einen Gegenstand kommuniziert werden können, wird das Objekt simultan von verschiedenen Seiten visualisiert. Für eine gestalterhaltende und gleichzeitig detaillierte Darstellung eines einzelnen Einrichtungsgegenstandes wird, aufgrund projektionsbedingter Verzerrungen in zentralperspektivischen Abbildern, auf den Einsatz einer konventionellen monoperspektivischen Darstellung verzichtet. Stattdessen wird für eine gestalterhaltende, verzerrungsfreie Darstellung ein multiprojektives Abbildungsverfahren eingesetzt. Hierbei dient das *Entwurfsmuster Sichtweise* als Vorlage für den Entwurf eines Kameramodells zur Erzeugung einer bildraumbasierten multiperspektivischen Darstellung. Die Umsetzung auf Basis des Entwurfsmusters wird in Abbildung 50 am Beispiel des Sessels, der sich im Fokus der Überblicksansicht befindet veranschaulicht. Für die Visualisierung des Möbelstückes werden drei Einzelkameras miteinander verknüpft. Dieser Verbund von Kameras mit jeweils zentralprojektiver Abbildungsvorschrift besteht aus einer Systemkamera sowie zwei Objektkameras. Die Systemkamera dient der Abbildung des Sessels in einer *Frontalansicht*. Dazu werden der virtuelle Ausstellungsraum und das Möbelstück gerendert, während alle weiteren Gegenstände von der Bildsynthese ausgeschlossen sind. Die Abbilder des Sessels aus einer *seitlichen* sowie einer *rückseitigen Perspektive* werden durch die separaten Objektkameras erzeugt. Hierzu besitzen diese gegenüber der Szenenkamera eine veränderte Position und Ausrichtung, während die weiteren Kameraparameter übernommen werden, um ein konsistentes Gesamtbild zu erzeugen. Die Zusammenführung der Einzelabbilder erfolgt durch eine Bildebenenkomposition, wodurch der Bildraum als eine *Multiprojektion* erzeugt wird. Die Realisierung einer nutzerabhängigen Hinein- und Herausbewegung zur Exploration des Gegenstandes in der multiperspektivischen Detailansicht wird durch eine gleichmäßige *Tiefentranslation* des gesamten Kameramodells in Richtung des Einrichtungsgegenstandes erreicht.

Zusätzlich zur Festlegung des sichtinternen Verhaltens ist der konkrete Entwurf für die sichtübergreifende Veränderung der 3D-Benutzerschnittstelle notwendig. Hierbei wird eine Kopplung der Sichten vollzogen, um das zentrierte Objekt der Überblicksdarstellung in der Detailansicht zu visualisieren. Diese *Sichtanpassung* wird auf Basis eines Modifikators umgesetzt, welcher kontinuierlich den Wert des Kamerasichtvektors der Überblicksdarstellung an den Sichtvektor der Szenenkamera in der Detailansicht überträgt. Dadurch wird das Kameramodell der Detailansicht automatisch auf das jeweils fokussierte Objekt ausgerichtet. Im Resultat wird dadurch eine automatische und kontinuierliche Anpassung der Detailansicht in Abhängigkeit von der Nutzerinteraktion in der Überblicksansicht umgesetzt.



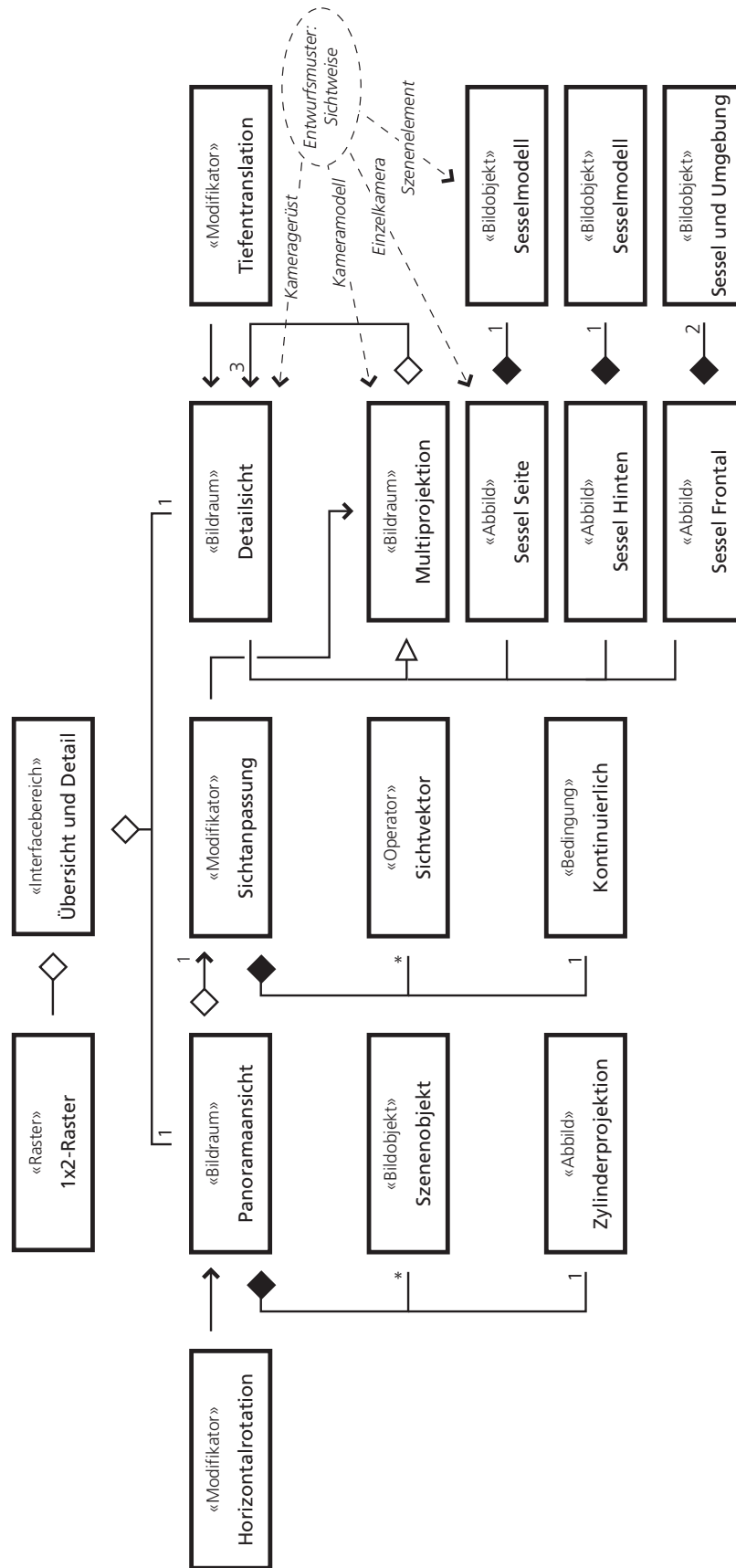


Abbildung 50: Das konkrete Interfacedesign für das Übersicht-und-Detail-Szenario auf Basis des MosaiK:Konkret-Profils und unter Einbeziehung des Entwurfsmusters Sichtweise.

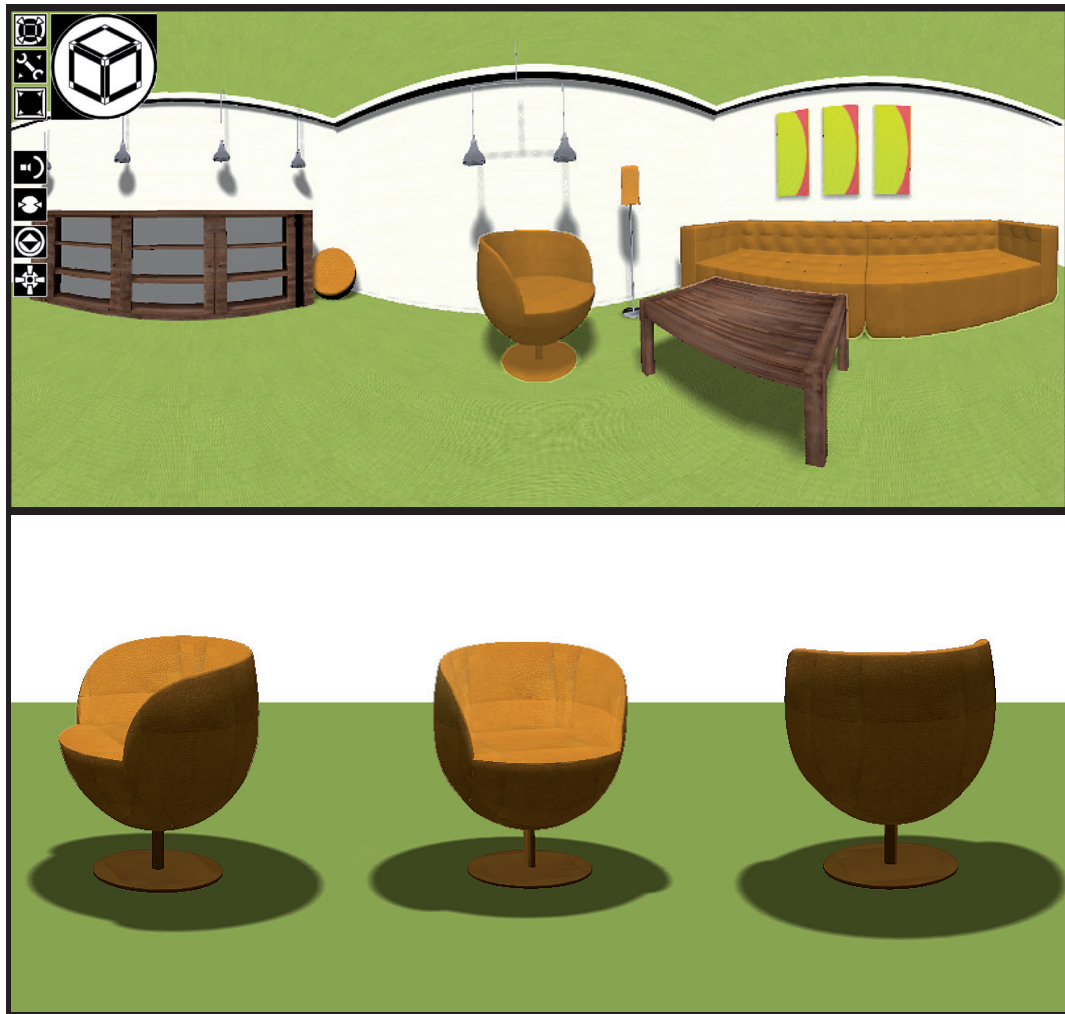


Abbildung 51: Das finale Interface des Übersicht-und-Detail-Szenarios im Bildsprache LiveLab

### Der finale Interfaceentwurf

Basierend auf dem konkreten 3D-Interfacedesign wird in der Implementierungsphase die finale 3D-Benutzerschnittstelle umgesetzt. Dies kann einerseits durch die Beschreibung des Interface in einem MosaK-Graph Instanzdokument auf Basis der Beschreibungssprache MosaK:Dialog erfolgen. Andererseits ist die Verwirklichung des Fallbeispiels durch einen Kompositions- und Konfigurationsprozess in der Arbeitsumgebung Bildsprache LiveLab möglich. Das Resultat, die finale 3D-Benutzerschnittstelle ist in Abbildung 51 dargestellt. Die Abbildung zeigt den Ausstellungsraum des fiktiven Einrichtungshauses. Die Überblicksansicht (oben) wird im Bildsprache LiveLab durch die zylindrische Panoramadarstellung mit einem Sichtbereich von 270 Winkelgrad verwirklicht. Im Zentrum der Visualisierung ist der Sessel dargestellt, welcher bei der Beschreibung des Kameramodells im konkreten 3D-Interfaceentwurf als Beispielobjekt dient. Durch die Kopplung der zwei Sichten ist das Möbelstück ebenso in der Detailansicht (unten) abgebildet. Der Sessel wird für eine detaillierte Darbietung des äußeren Erscheinungsbildes simultan von drei Seiten visualisiert.

```

1| ...
2| <!-- Definition des Panelknotens -->
3| <panel DEF="DetailView" rasterRowPos="1" rasterColPos="0" rasterRowSpan="1" rasterColSpan="1"
4|   panelRows="1" panelColumns="1" bgColor="black">
5|   <!-- Definition des Bildraumknotens -->
6|   <image panelRowPos="1" panelColPos="0" panelRowSpan="0" panelColSpan="1" imageRows="1"
7|     imageColumns="3" imageType="multiple" projCount="3">
8|     <!-- Definition der Kamera für den zentrierten Gegenstand und den umgebenden Objektraum -->
9|     <camera DEF="DetailMainCamera">
10|       <projection type="linear"/>
11|       <vector position="0,-5,0" horizontalFOV="60" up="z" direction="0,0,1"/>
12|       <principlePoint left="0.5" right="0.5" top="0.5" bottom="0.5" depth="1"/>
13|       <viewport imageRowPos="0" imageColPos="1" imageRowSpan="1" imageColSpan="1"/>
14|     </camera>
15|     <!-- Definition der Kamera für die Rückansicht -->
16|     <camera DEF="BackCam">
17|       <projection type="linear"/>
18|       <vector position="0,5,0" horizontalFOV="60" up="z" direction="0,0,-1"/>
19|       <principlePoint left="0.5" right="0.5" top="0.5" bottom="0.5" depth="1"/>
20|       <viewport imageRowPos="0" imageColPos="2" imageRowSpan="1" imageColSpan="1"/>
21|     </camera>
22|     <!-- Definition der Kamera für die Seitenansicht -->
23|     <camera DEF="LeftCam">
24|       <projection type="linear"/>
25|       <vector position="5,0,0" horizontalFOV="60" up="z" direction="1,0,0"/>
26|       <principlePoint left="0.5" right="0.5" top="0.5" bottom="0.5" depth="1"/>
27|       <viewport imageRowPos="0" imageColPos="0" imageRowSpan="1" imageColSpan="1"/>
28|     </camera>
29|   </image>
30| </panel>
31| ...
32| <!-- Definition eines Verhaltens zur Koordination der Sichten -->
33| <modifier DEF="CoordinatedViews" type="AttributeLink">
34|   <operator fromNode="OverviewCamera" fromComponent="direction"
35|     toNode="DetailMainCamera" toComponent="direction" relative="false" offset="0,0,3"/>
36| </modifier>
37| ...

```

**Listing 3:** Das Übersicht-und-Detail-Szenario als MosalK-Graph (Auszug) umfasst die Definition der Szenenkamera, der Objektkameras für die Detailansicht und deren strukturelle Einordnung in den MosalK-Graph sowie die Definition von Modifikatoren zur Sichtenkopplung und Objektzuweisung.

Das Listing 3 zeigt Auszüge aus dem MosalK-Graph-Instanzdokument des Fallbeispiels in einer gekürzten Form. Der Quelltextausschnitt zeigt die Kameradefinition für die Detailansicht und deren Gruppierung als Kinder eines Bildraumknotens. Durch die Paneldefinition als Elternknoten des Bildraums wird ferner dessen Position auf dem Interfaceraster definiert. Weiterhin umfasst der Quelltext die Definition eines Modifikatorknotens, der wiederum einen Operator besitzt. Der Modifikator beschreibt die Verknüpfung von Attributen (*AttributeLink*) und legt die Kopplung von Detail- und Überblickssicht auf Basis des Kamerasichtvektors fest.

## 6.3 Anwendungsszenarien im Kontext wissenschaftlicher Visualisierungen

Im Folgenden werden zur Überprüfung der Praktikabilität des MosalK-Vorgehens und zur Veranschaulichung des Gestaltungspotenzials interaktiver 3D-Benutzerschnittstellen zwei Anwendungsszenarien beschrieben. Durch die Beschreibung von Szenarien, denen jeweils spezifische Visualisierungsziele zugrunde gelegt sind, wird die Anwendbarkeit komponierter

Abbildungsverfahren in 3D-Benutzerschnittstellen mithilfe der interaktiven Arbeitsumgebung Bildsprache LiveLab verdeutlicht.

Die Szenarien entstammen verschiedenen Wissenschaftsbereichen, deren zugrunde liegende Daten vielfältige Charakteristiken aufweisen. Im Allgemeinen können die Daten wissenschaftlicher Visualisierungen einerseits durch Messgeräte erfasst oder auf Basis mathematischer Modelle erhoben werden. Eine dritte Gruppe (vgl.[SCHUMANN & MÜLLER 2000, S. 27–28]) bilden vom Menschen entworfene Daten, die zum Zeitpunkt der Erzeugung keine real existierende Entsprechung abbilden. Da diese dritte Kategorie bereits durch das Fallbeispiel der Produktvisualisierung beleuchtet wird, werden nachfolgend zwei Anwendungsszenarien vorgestellt, deren grundlegende Daten der realen sowie der theoretischen Welt [SCHUMANN & MÜLLER 2000, S. 27–28] entstammen. Die Darlegung von Anwendungsszenarien, die auf unterschiedlichen Datenquellen beruhen, soll die Vielgestaltigkeit des Anwendungsbereiches wissenschaftlicher Visualisierungen hervorheben, in denen der Einsatz von interaktiven 3D-Benutzerschnittstellen möglich und zweckdienlich ist. Um ferner die Aktualität und die Relevanz des Anwendungskontextes in Bezug auf die Anforderungen an ein interaktives System sicherzustellen, basieren die zwei Szenarien auf aktuellen Forschungsthemen der Technischen Universität Dresden.

Die Unfallforschung erhebt und visualisiert Daten, die der Kategorie der realen Welt zuzuordnen sind. Dieses erste Anwendungsszenario gibt einen Einblick in die Forschung sensorbasierter Sicherheitssysteme der Technischen Universität Dresden und wird im Unterabschnitt 6.3.1 beschrieben. Der Wissenschaftsbereich der Systembiologie bedient sich mathematischer Modelle zur Beschreibung biologischer Vorgänge und deren Daten ausgehend von einer theoretischen Datenquelle im Kontext wissenschaftlicher Visualisierungen. Das Szenario stellt das Projekt „Virtual Liver“ vor, an dem sich die Technische Universität Dresden zusammen mit 70 weiteren Forschungseinrichtungen beteiligt. Ein Aspekt dieses Forschungsvorhabens wird im Abschnitt 6.3.2 vorgestellt.

### 6.3.1 Anwendungsszenario: Unfallforschung

Die Unfallforschung der Technischen Universität Dresden befasst sich mit der Erhebung, Auswertung und Dokumentation von Realunfalldaten im Straßenverkehr. Mithilfe erhobener Daten werden Weiterentwicklungen im Bereich der Sicherheitstechnik von Fahrzeugen, Verkehrsteilnehmern und Verkehrsanlagen vollzogen. Die Forschung und Entwicklung erfolgt mit dem Ziel, eine Verringerung von materiellen Schäden sowie Personenschäden im öffentlichen Straßenverkehr zu erreichen. Ein Schwerpunkt innerhalb der Unfallforschung stellt dabei die Entwicklung sensorbasierter Sicherheitssysteme dar. Die Systeme reagieren in der Unfalleinleitungsphase, um den tatsächlichen Zusammenstoß zu verhindern oder den entstehenden Schaden zu reduzieren. Zur Untersuchung und Analyse von Unfalleinleitungsphasen werden Simulationen durchgeführt. Diese werden auf Basis erhobener Daten realer Unfallorte und der GIDAS<sup>64</sup> Datenbank realisiert (vgl. [ERBSMEHL & HANNAWALD 2008, S. 2]). Die Datenbasis ermöglicht eine Nachbildung der Unfalleinlaufbewegungen aller Beteiligten. Dies

<sup>64</sup> Abkürzung für German In-Depth Accident Study

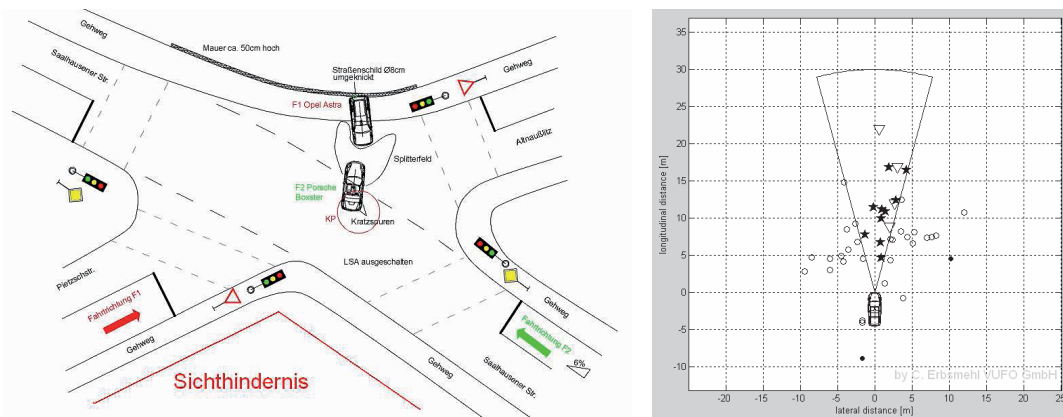


Abbildung 52: Wissenschaftliche Visualisierungen aus dem Bereich der Unfallforschung: Skizze eines Unfalls von zwei Personenkraftwagen (links) und ein pre crash scatter plot zur schematische Darstellung der Detektion eines Sensorsystems (rechts) (aus [ERBSMEHL & HANNAWALD 2008])

wird erreicht, indem im Zuge einer Unfalldokumentation eine Fallakte mit Unfallfotos, Verletzungsübersichten, Beschädigungsmustern und Unfallskizze angefertigt wird. In dieser Unfallskizze ist neben essenziellen Informationen zu den Unfallbeteiligten, auch die Position von Sichthindernissen, Fahrlinien und Fahrbahnbegrenzungen festgehalten. Die Abbildung 52 zeigt eine solche Unfallskizze der Unfallforschung der Technischen Universität Dresden auf Basis eines realen Unfallhergangs. Die verfügbaren Informationen werden für die Simulationen von Unfallhergängen verwendet. Diese basieren auf den erhobenen geometrischen Positionen, den gefahrenen Geschwindigkeiten und den eingeleiteten Manövern der Unfallbeteiligten. Die Auswertung der Simulationen erfolgt auf Basis von *pre crash scatter plots*. Durch die grafische Veranschaulichung der zur Verfügung stehenden Daten können Sichtbarkeitsbedingungen sowie die Wirksamkeit eines bestimmten Sensorsystems am Einzelfall ausgewertet werden (vgl. [ERBSMEHL & HANNAWALD 2008]). Die Abbildung 52 zeigt die Ergebnisdarstellung eines Simulationsprozesses als *pre crash scatter plots*. Durch die Darstellung werden die Eigenschaften und die Funktionsweise von Sensorsystemen an realen Unfallhergängen visuell überprüfbar, da Informationen und Zusammenhänge sowie Wechselwirkungen zwischen den Beteiligten im Unfallgeschehen aufgezeigt werden. Auf Basis der Auswertungen können sensorbasierte Sicherheitssysteme entworfen und Effizienzabschätzungen zu aktiven Sicherheitssystemen durchgeführt werden.

Der Einsatz von Visualisierungen bei der Analyse und der Auswertung von realen Unfalldaten veranschaulicht die Relevanz von grafischen Darstellungen bei der Analyse und Erforschung von Sensortechniken und Sicherheitssystemen. Darüber hinaus wird deutlich, dass in den Analyse-, Simulations- und Auswertungsansichten verschiedene Darstellungsarten verwendet werden, um wesentliche Informationen im jeweiligen Kontext herauszustellen. Dabei sind die Darstellungen überwiegend statisch und bilden nur eine Augenblicksaufnahme des Unfallgeschehens ab. Darüber hinaus bieten diese nur eine begrenzte räumliche Darstellung, wodurch beispielsweise in den *pre crash scatter plots* die horizontale Detektion von Hindernissen identifizierbar, die vertikale Detektion jedoch nicht in einem gemeinsamen Abbild dargestellt wird (siehe Abbildung 52). Die Veranschaulichung eines Sensorsimulationsprozesses in einer

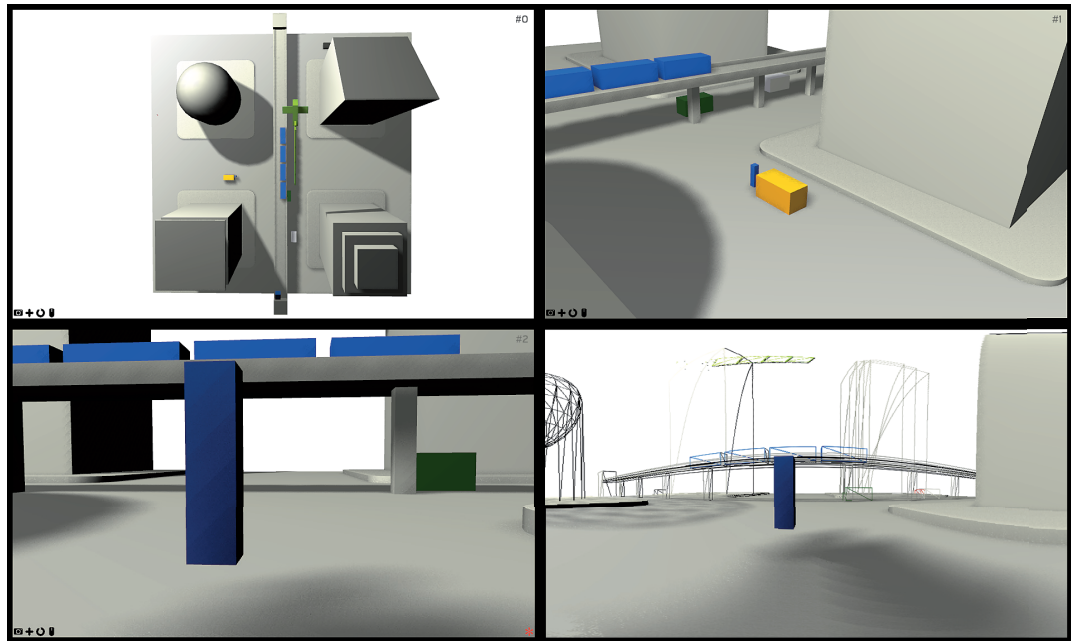


Abbildung 53: 3D-Interface mit der Darstellung eines Unfalls auf Basis multipler Sichten. Die Sichten zeigen eine Überblicksdarstellung (links oben), eine perspektivische Sicht auf die Unfallsituation (oben rechts), die Situation aus der Fahrersicht (unten links) und die Sichtbarkeitsbereiche (texturierte Bereiche) eines Sensorsystems (unten rechts) (Quelle: Bildsprache LiveLab [AG-TV]).

interaktiven 3D-Benutzerschnittstelle ermöglicht hingegen die Darstellung des räumlichen Sensorsichtkörpers in einem Abbild. Dies unterstützt das Verständnis für den konkreten Sachverhalt durch die dreidimensionale Darstellung des nachempfundenen realen Ereignisses. Ferner kann in einer interaktiven Anwendung die gesamte Unfalleinleitungsphase in einer dynamischen Umgebung dargestellt werden und dadurch die Faktoren in einem zeitlichen Zusammenhang betrachtet werden. Aufgrund der Komplexität einer Unfallsimulation und der Menge an Faktoren bietet eine visuelle Trennung des Simulationsvorganges in mehrere Sichten eine Verringerung der jeweiligen Komplexität. Durch multiple Sichten auf einen Datenbestand werden die Disparitäten und die Zusammenhänge des Unfallherganges erkennbar. Darüber hinaus können multiple koordinierte Sichten auf einen Datenbestand dem Betrachter ein besseres Verständnis für die Einflüsse und die Folgen einer Parameteränderung innerhalb der Unfallsimulation bieten. Eine mögliche Umsetzung einer 3D-Benutzerschnittstelle zeigt Abbildung 53, die mithilfe BiLL-Anwendung unter Einbeziehung der Komponenten für multiple Sichten und bildraumbasierte Multiperspektiven erzeugt wird. Die verschiedenen Sichten zeigen eine Unfallsituation. Neben einer Überblicksdarstellung und einer perspektivischen Sicht auf den Unfallort werden zusätzlich die Fahrersicht und der Sensorsichtbereich dargestellt. Der simulierte Sensor verfügt über einen Sichtbereich von 175 Winkelgrad und wird durch eine Streifenkamera, bestehend aus Einzelkameras verbildlicht.

Durch die Charakteristiken eines interaktiven Systems ist es möglich, den Unfallhergang im Verlauf zu visualisieren und darüber hinaus interaktiv in diesen einzugreifen. Durch die Kopplung der Sichten hat eine Veränderung einen sofortigen Einfluss auf die weiteren



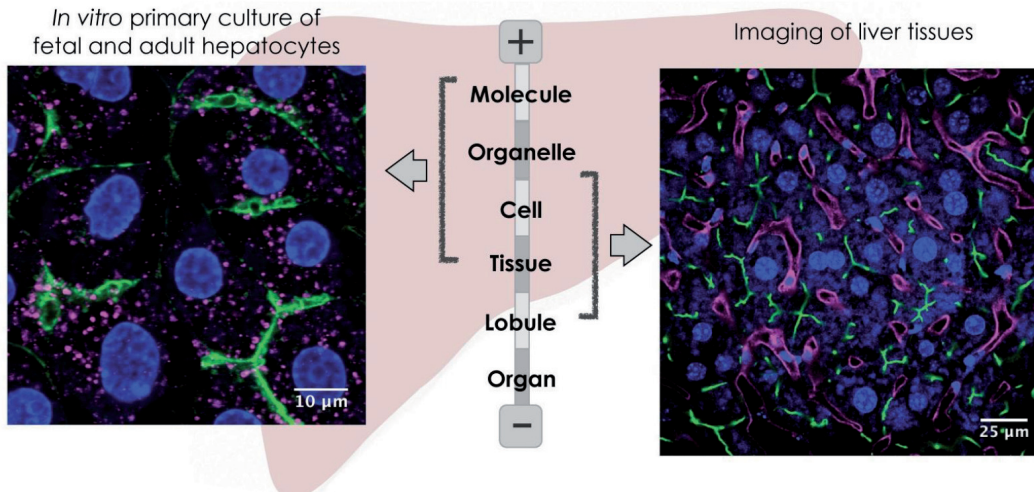


Abbildung 54: Darstellung der Leber auf zellulärer Ebene und auf Gewebeebene aus [NONAKA u. A. 2011]

Sichten. Dadurch können beispielsweise Detailansichten im Kontext der Gesamtsituation betrachtet oder durch eine gekoppelte Fahrer- und Sensorsicht Sichtbarkeitsvergleiche und –prüfungen durchgeführt werden.

### 6.3.2 Anwendungsszenario: Systembiologie

Die Systembiologie ist ein Forschungszweig der Biowissenschaften, welcher sich der Erforschung biologischer Prozesse widmet. Die dynamischen Wechselwirkungen zwischen den Bestandteilen eines biologischen Systems werden in unterschiedlichen interdisziplinären Verfahren mit dem Ziel analysiert, das Verhalten des Systems als Ganzes zu verstehen und Vorhersagen zu ermöglichen. In einem iterativen Prozess unter Einbeziehung von Laborexperimenten und computergestützter Modellierungen werden mathematische Konzepte auf biologische Systeme angewendet. Mit dem „Virtual Liver Network“ haben sich Forscher der Technischen Universität Dresden und einer Vielzahl weiterer Forschungseinrichtungen aus den Bereichen Biologie, Mathematik, Physik und Medizin ein gemeinsames Ziel gesetzt. Mit der „Virtuellen Leber“ soll ein dynamisches mathematisches Modell entwickelt werden, das die Physiologie, die Morphologie und die Funktion der menschlichen Leber beschreibt. Das übergeordnete Ziel dieses Forschungsvorhabens ist es, Organtransplantationen sicherer zu machen sowie neue und effektivere Medikamente zu entwickeln. Dabei werden quantitative Daten aus allen Organisationsstufen der Leber in das Modell integriert. Skalaübergreifende Darstellungen der Leberphysiologie vom Zell- bis zum Organlevel dienen dem Verständnis für die Prozesse und die Wechselwirkungen innerhalb der (virtuellen) Leber. Dafür bedarf es unterschiedlicher Daten über Vorgänge und Abläufe, die auf unterschiedlichen räumlichen und zeitlichen Skalen beschrieben werden. Die visuelle Ausgabe der Modellierungen erfolgt derzeit vorwiegend für die jeweilige Betrachtungsebene. Die Abbildung 54 veranschaulicht einerseits eine Aufnahme des Lebergewebes, andererseits in einer vergrößerten Darstellung die Leberzellen mit verschiedenen Zellorganellen. Die Verknüpfung der beiden Ebenen erfolgt



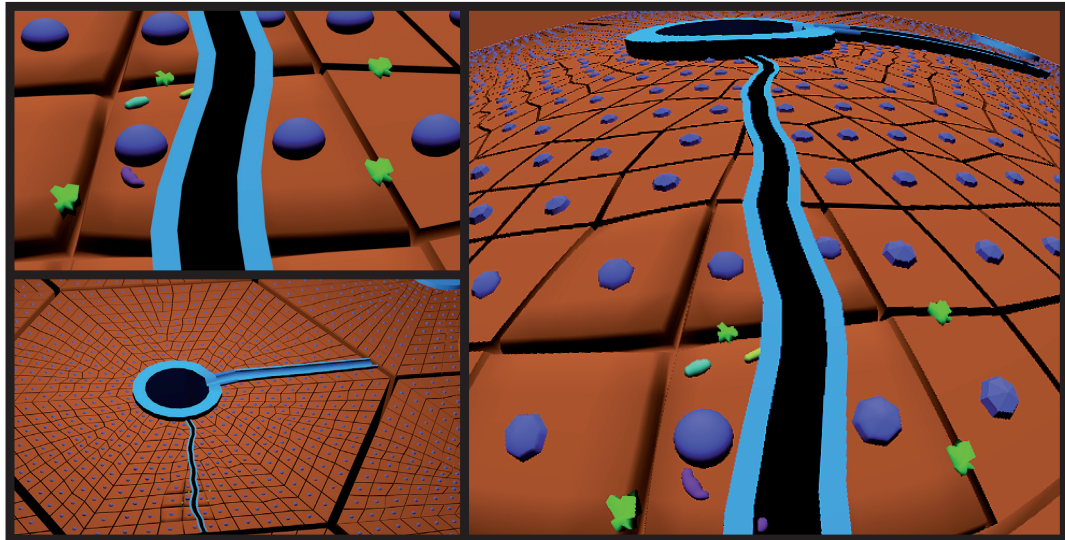


Abbildung 55: Modellhafte Nachbildung einer Molekülstruktur der Leber in einer Übersichtsdarstellung (links unten), in einer Detailsicht (links oben) sowie die Zusammenführung der zwei Ebenen in einer Fischaugendarstellung (rechts) (Quelle: Bildsprache LiveLab [AG-TV])

lediglich schematisch über eine Bereichseinteilung. Folglich müssen die charakteristischen Merkmale einer Ebene zunächst identifiziert werden, um in einem darauf folgenden Schritt die Ebenen funktional miteinander verbinden zu können. Nur durch die semantische Verknüpfung der Ebenen ist die Möglichkeit zu einem allgemeinen Verständnis der Leberfunktionalität gewährleistet. Eine alternative Darstellungsweise unter Verwendung von nichtlinearen Darstellungsverfahren wird am Beispiel von Krebszellen in [BEHRENS U. A. 2010] gezeigt. Durch eine Fokus-und-Kontext-Darstellung ergibt sich die Möglichkeit einer bildstrukturellen und gleichzeitig semantischen Verknüpfung der Ebenen in einem statischen Abbild. Ein vergleichbarer Ansatz bietet sich auch für 3D-Benutzerschnittstellen interaktiver Anwendungen im „Virtual Liver Network“ an. Mit dieser Darstellungstechnik können die regulatorischen Prozesse des mathematischen Modells der virtuellen Leber über mehrere Ebenen hinweg verbildlicht und darüber hinaus interaktiv manipuliert werden. Die Abbildung 55 zeigt das Modell mithilfe einer Fokus-und-Kontext-Darstellung. Während die zwei Einzelsichten auf der linken Seite keinen direkten Zusammenhang aufweisen und dieser als zusätzliche Information der Darstellung beigefügt werden muss (siehe Abbildung 54), bietet eine krummlinige Darstellung die Möglichkeit, die zwei Ebenen in einem Abbild zusammenzuführen. Durch den interaktiven Charakter des 3D-Interface im Bildsprache LiveLab, unter Verwendung der Visualisierungskomponente kurvilinearere Projektionen, kann der Bildraum, in Abhängigkeit vom Visualisierungsziel, in einem interaktiven Prozess verändert werden.

## 6.4 Fazit aus den Betrachtungen

Die Beispiele beschreiben die Visualisierung von Modellen, Konzepten und Daten im Kontext der Produktvisualisierung, der Unfallforschung und der Systembiologie. Dabei werden konkrete Aspekte des jeweiligen Bereiches aufgegriffen und für die Veranschaulichung der

Umsetzbarkeit in einer 3D-Benutzerschnittstelle verwendet. Es konnte gezeigt werden, dass die visuelle Repräsentation von Daten in einer interaktiven 3D-Anwendung ein essenzielles Hilfsmittel für die Exploration, Analyse und Präsentation von Daten darstellt. Insbesondere wird der zusätzliche Nutzen durch den Einsatz nichtlinearer Darstellungsverfahren in koordinierten multiplen Sichten für die Präsentation von Ergebnissen und Sachverhalten sowie für die visuelle Analyse einer gegebenen Datenmenge in interaktiven Systemen verdeutlicht. Dabei erfolgte die Entwicklung der 3D-Benutzerschnittstellen unter Einsatz des MosalK-Vorgehens und die Verwirklichung im Bildsprache LiveLab. Durch die Visualisierungsbausteine der BiLL-Architektur konnten die Benutzerschnittstellen in der Echtzeitumgebung zielgerichtet konfiguriert und für den konkreten Anwendungskontext spezifiziert und prototypisch eingesetzt werden.

## 7 Schlussbetrachtung

Der Forschungsbereich der Gestaltung und des Entwurfes interaktiver 3D-Benutzerschnittstellen – vor allem unter Berücksichtigung nicht-fotorealistischer Abbildungsverfahren – wurde in dieser Arbeit detailliert betrachtet. Dabei ist deutlich geworden, dass konzeptionelle und methodische Unzulänglichkeiten den erfolgreichen Einsatz von unkonventionellen linearen sowie nichtlinearen Darstellungsformen in 3D-Anwendungen bisher einschränken. In dieser Arbeit wurden daher Darstellungstechniken analysiert und systematisiert, die durch den computergrafischen Projektionsvorgang erzeugt und für die Verwirklichung von Visualisierungszielen eingesetzt werden können. Ferner erfolgte eine Einführung von Gestaltungshinweisen für den spezifischen Einsatz von nicht-fotorealistischen Visualisierungsverfahren in 3D-Benutzerschnittstellen und die Einführung einer Vorgehensweise, durch welche die systematisierten Visualisierungsformen in einem modellbasierten und ferner entwurfsmuster-gestützten Entwurfsprozess zur Entwicklung interaktiver 3D-Interfaces eingesetzt werden können.

### 7.1 Zusammenfassung

#### Kapitel 1: Einleitung

Im einführenden Kapitel wurden die Ausgangssituation und die Motivation zu dieser Arbeit erläutert. Dabei wird den nicht-fotorealistischen Abbildungsverfahren eine wichtige Rolle bei der Gestaltung künftiger 3D-Benutzerschnittstellen attestiert, da aufgrund der Vielgestaltigkeit der Darstellungsformen eine Flexibilität in der visuellen Informationsdarbietung und der Veranschaulichung von 3D-Szenen erreicht sowie die Etablierung eines kontextabhängigen Nutzer-Bild-Dialogs verwirklicht werden können. Es werden ebenso zahlreiche Herausforderungen offenkundig, welche die Konzeption, den Entwurf und den Einsatz von nicht-fotorealistischen Abbildungsverfahren in Benutzerschnittstellen interaktiver 3D-Systeme erschweren. Ausgehend von einer Vision wurden die Thesen, die Ziele und der strukturelle Aufbau der Arbeit dargestellt, sowie die Abgrenzung zu verschiedenen Fachgebieten vorgenommen.

#### Kapitel 2: Grundlagen interaktiver 3D-Anwendungen

Das zweite Kapitel führte in die Thematik der Arbeit ein, indem eine Charakterisierung computergrafischer 3D-Darstellungen erfolgte und ferner die Merkmale interaktiver 3D-Benutzerschnittstellen aufgezeigt und erläutert wurden. Der weitverbreitete Einsatz interaktiver Anwendungen konnte ferner als eine Motivation für die verschiedenen Forschungsarbeiten in den Bereichen der modellbasierten und entwurfsmuster-gestützten Benutzerschnittstellenentwicklung identifiziert werden. In diesem Zusammenhang wurden

verschiedene Verfahrensweisen für den Interfaceentwurf, die sich durch die unterschiedlichen Zielsetzungen der beteiligten Personengruppen ergeben, aufgezeigt. Eine Reflexion der ineinandergreifenden gestalterischen, technischen und methodischen Herausforderungen in der Benutzerschnittstellenentwicklung rundete das Kapitel ab.

### **Kapitel 3: Abbildungsverfahren für interaktive 3D-Anwendungen**

Im dritten Kapitel erfolgte eine detaillierte Betrachtung linearer und nichtlinearer Abbildungsverfahren im Hinblick auf deren Einsatz in interaktiven 3D-Anwendungen. Dazu wurde zunächst ein historischer Abriss über 3D-Benutzerschnittstellen gegeben, Anwendungsdomänen aufgezeigt sowie Forschungsprojekte und Beispielanwendungen vorgestellt. Eine Analyse von 3D-Benutzerschnittstellen zeigte divergierende Argumente in Bezug auf deren Einsatz im Anwendungskontext auf. In der Folge diente eine Betrachtung der bildstrukturellen Eigenschaften von 3D-Darstellungen, die durch eine weiterführende Analyse von Gemälden der neuzeitlichen Malerei ergänzt wurde, die Vielfalt von Darstellungsformen, die für einen intendierten Nutzer-Bild-Dialog eingesetzt werden können, aufzuzeigen. Für eine Einbindung der Gestaltungsmöglichkeiten in den 3D-Interfaceentwurf wurde die computergrafische Kamera als Gestaltungswerkzeug identifiziert und deren Flexibilität durch die Veranschaulichung des Aufbaus und der Modifikationsmöglichkeiten aufgezeigt. Motiviert durch die Variationen den computergrafischen Projektionsvorgang zu beeinflussen, wurden Gestaltungshinweise für 3D-Benutzerschnittstellen formuliert, die eine allgemeingültige Unterstützung im Umgang mit nicht-fotorealistischen 3D-Interfaces bieten. Ferner erfolgte aus der Analyse einer großen Anzahl wissenschaftlicher Arbeiten zu computergrafischen Abbildungsverfahren die Entwicklung einer Systematisierung von Darstellungstechniken im Hinblick auf die Verwirklichung von Visualisierungszielen in interaktiven 3D-Anwendungen.

### **Kapitel 4: BiLL: Ein Framework für den 3D-Interfaceentwurf**

Nach den theoretischen und analytischen Ausführungen des dritten Kapitels beschreibt das vierte Kapitel das Bildsprache LiveLab als komponentenbasierte Arbeitsumgebung für die technische Realisierung der Abbildungsverfahren in einer interaktiven Anwendung. Zunächst wurden hierzu die Anforderungen, die an eine geeignete Architektur, an integrative Bausteine und an die Funktionsweise als Arbeitsumgebung gestellt werden, aufgeführt. Die Ausführungen verdeutlichten, dass etablierte Software-Komponententechnologien in Verbindung mit einer Grafik-Engine und einem GUI-Toolkit einen zweckmäßigen Lösungsansatz für die interaktive 3D-Anwendung darstellen. In der Folge wurden bestehende Technologien hinsichtlich der Entwicklung einer Arbeitsumgebung betrachtet und der Einsatz von OSGi, OpenSceneGraph und wxWidgets als Basis des Bildsprache LiveLab erläutert. Die Verfügbarkeit von linearen und nichtlinearen Abbildungsverfahren in der interaktiven 3D-Anwendung wird durch die Integration der Verfahren als Visualisierungsbausteine in die serviceorientierte Komponentenarchitektur erreicht. Dies wurde durch eine kompakte Vorstellung der BiLL-Architektur erreicht und die Wirkungsweise der Arbeitsumgebung anhand des serviceorientierten Zusammenwirkens der Visualisierungsbausteine zur Verwirklichung von 3D-Interfaces erläutert. Weiterhin erfolgten Ausführungen zu den unterschiedlichen Aufgaben,

Resultaten und Werkzeugen in Bezug auf die Weiterentwicklung und den interdisziplinären Einsatz der Bildsprache LiveLab beim Entwurf interaktiver 3D-Benutzerschnittstellen.

## Kapitel 5: MosalK: Modellbasierte 3D-Interfacekomposition

Für den gezielten Einsatz des aufgezeigten Repertoires an Darstellungsweisen im Entwurfsprozess von 3D-Interfaces wurde im fünften Kapitel das modellgestützte Vorgehen MosalK entwickelt und detailliert beschrieben. Hierbei standen zunächst die Herausforderungen und ferner system- sowie benutzerzentrierte Vorgehensweisen im Entwurfsprozess mit deren Modellen und Beschreibungssprachen im Mittelpunkt der Betrachtung. Darauf aufbauend erfolgte die Entwicklung des MosalK-Vorgehens, wobei die eingesetzten Sprachen und Notationen herausgearbeitet wurden, wodurch die notwendige Präzision im modellbasierten Vorgehen sichergestellt werden konnte. Mit dem MosalK-Vorgehen, das eine Überführung einer Anforderungsspezifikation in eine 3D-Benutzerschnittstellenbeschreibung vollzieht, wurden einerseits die UML-Profile MosalK:Abstrakt und MosalK:Konkret für die Modellierung, andererseits die XML-basierte, deklarative Auszeichnungssprache MosalK:Dialog für die Beschreibung von 3D-Benutzerschnittstellen eingeführt. Durch eine Aufteilung des Entwurfsprozesses in drei Phasen wird eine strukturierte und aufeinander aufbauende Entwicklung von 3D-Benutzerschnittstellen sichergestellt, in deren Folge die verschiedenen Modelle und Sprachen gezielt in der jeweiligen Phase eingesetzt werden können. Vorgehensbegleitend wurden für die Etablierung, die Anwendung und die Wiederverwendung von Gestaltungswissen Entwurfsmuster eingeführt, um die Erfüllung softwareergonomischer Anforderungen im Entwurfsprozess zu unterstützen. Mit der Verbindung von einem modellbasierten Vorgehen und einer Entwurfsmustermenge wird, mit dem Ziel der Gestaltung interaktiver Systeme, die Beschreibung eines Datenbildes um Werkzeuge für die Entwicklung eines Dialogbildes ergänzt.

## Kapitel 6: MosalK und BiLL im Anwendungskontext

Die Anwendbarkeit der Ergebnisse dieser Arbeit wurde im sechsten Kapitel durch den Einsatz des MosalK-Vorgehens auf Basis von Anwendungsszenarien verdeutlicht. Auf Grundlage eines Fallbeispiels aus dem Bereich der Produktvisualisierung wurde das Durchlaufen der einzelnen Schritte im modellbasierten Vorgehen an einem konkreten Szenario aufgezeigt. Die Realisierung des finalen Benutzerschnittstellenentwurfes wurde im Bildsprache LiveLab, basierend auf den Visualisierungsbausteinen der Arbeitsumgebung und einem MosalK-Graph-Instanzdokument, vollzogen. Dazu wurde das Bildsprache LiveLab zunächst aus Anwendersicht vorgestellt, das Erscheinungsbild der Anwendung, darunter das Editor- und das Viewer-Fenster sowie der Dialog zur Verwaltung von Visualisierungsbausteinen, beschrieben. Weiterhin erfolgte die Darlegung von Szenarien aus den Bereichen Systembiologie und Unfallforschung, durch die der Einsatz von linearen und nichtlinearen Abbildungsverfahren in 3D-Benutzerschnittstellen einer interaktiven Anwendung für wissenschaftliche Visualisierungen verdeutlicht wird.

## 7.2 Fazit

Die in der Einleitung dieser Arbeit aufgestellten Annahmen dienen primär als Leitlinien und Motivation im wissenschaftlichen Diskurs. Diese lassen sich erst durch umfassende und teils langfristige Forschungsarbeiten abschließend bewerten. Dazu werden jedoch in dieser Arbeit wichtige Grundlagen gelegt. Für eine Reflexion der Arbeit können die in Unterabschnitt 1.2 gestellten Forschungsziele den Resultaten gegenübergestellt und deren Bezüge zu den Arbeitsthemen dargelegt werden.

Anhand der Kapitelzusammenfassungen wird deutlich, dass die Ziele der Arbeit erreicht wurden. Die vorgestellte Klassifikation computergrafischer Abbildungsverfahren ordnet die Bildräume hinsichtlich deren bildstruktureller Eigenschaften und in Bezug auf die Manipulationsmöglichkeiten der computergrafischen Kamera. Ferner wird durch die Systematisierung von Darstellungsverfahren im Hinblick auf mögliche Visualisierungsziele eine Einordnung bestehender Abbildungsverfahren, unter Berücksichtigung eines pragmatischen Charakters, im Sinne einer tatsächlichen Verwendbarkeit vorgenommen. In der Folge bietet die Systematisierung eine Grundlage, um 3D-Benutzerschnittstellen ausgehend vom zu erzeugenden Bildraum und dem potenziellen Anwender und nicht von der computergrafischen Kamera und deren Parameter im Objektraum zu konzipieren. Darüber hinaus trägt die Zusammenstellung computergrafischer Visualisierungsverfahren zur Bekanntmachung des verfügbaren Repertoires an Darstellungstechniken für den Entwurfsprozess interaktiven 3D-Anwendungen bei. Ein zielgerichteter Einsatz der Abbildungsverfahren erfolgt durch eine Komposition von Objektraumelementen und Teilabbildern im Bildraum sowie der Montage von Bildräumen in einer 3D-Benutzerschnittstelle. Die Gestaltungsvielfalt wird ferner durch eine variante Anordnung von Sichten und einen veränderlichen Aufbau von Bildräumen erreicht. Die vorgestellten Gestaltungshinweise für 3D-Benutzerschnittstellen bieten hierbei, bezogen auf den Einsatz im Interfaceentwurf, Vorgaben für die nutzer- und aufgabenzentrierte Gestaltung und Entwicklung. Ferner reflektieren die Hinweise die Notwendigkeit eines flexiblen Einsatzes der systematisierten Darstellungsverfahren zur Erzeugung kommunikationsunterstützender 3D-Interfaces. Die Realisierung von Abbildungsverfahren, als Visualisierungsbausteine in der serviceorientierte Komponentenarchitektur Bildsprache LiveLab, bietet die angestrebte Verfügbarkeit von Darstellungsweisen in einer Arbeits- und Experimentierumgebung. Die Architektur des Bildsprache LiveLab erleichtert einerseits die Entwicklung von neuen beziehungsweise modifizierten Darstellungstechniken. Andererseits unterstützt die Systemarchitektur, durch die Konfiguration, Assemblierung und das Zusammenwirken von Komponenten, den Einsatz von Visualisierungsverfahren für eine bildgestützte Interaktion in unterschiedlichen Anwendungskontexten. Ferner ermöglicht das Bildsprache LiveLab den interaktiven Prozess mit dem Anwender, indem das System, in Form von steuerbaren Bilderzeugungsautomatismen, der Mensch-Maschine-Kommunikation assistiert.

Die Anwendbarkeit der Darstellungsweisen in einem methodischen und konzeptionellen Entwurfsprozess, zur Entwicklung von Benutzerschnittstellen interaktiver 3D-Anwendungen, wird durch die modellbasierte Vorgehensweise Mosalk erreicht. Die Möglichkeit der strukturierten Entwicklung eines Dialogbildes für ein konkretes Datenbild, auf Basis der Modellierungssprache UML und der Auszeichnungssprache XML, wird durch die verschiedenen

Phasen, Abstraktionsstufen und Modelle innerhalb der Vorgehensweise erreicht. Das Mosalk-Vorgehen erlaubt die Definition von Einzelsichten und deren Verhalten sowie die Beschreibung und das Zusammenwirken von nicht-fotorealistischen Abbildungsverfahren in multiplen Sichten einer interaktiven 3D-Benutzerschnittstelle. Darüber hinaus werden durch die XML-basierte Beschreibung des Dialogbildes eine Wiederverwendbarkeit und eine plattformunabhängige Verfügbarkeit von 3D-Interfacebeschreibungen gewährleistet, wodurch aufgaben- und nutzeradaptierte Benutzerschnittstellen einem breiten Anwenderkreis zugänglich gemacht werden können. Gleichwie das Ziel der Entwicklung einer Vorgehensweise zur Unterstützung des methodischen Interfaceentwurfes wurde die Konzeption von Mustervorlagen erreicht. Mit den Entwurfsmustern werden bewährte Lösungen für den 3D-Interfaceentwurf bereitgestellt, die darüber hinaus einen Wissenstransfer bewirken und infolgedessen die softwareergonomische Qualität eines 3D-Interface anwendungsübergreifend sicherstellen können. Durch die Bereitstellung der Mustervorlagen, die in dem methodischen Vorgehensmodell Anwendung finden, werden somit Designer und Entwickler in die Lage versetzt, in einem interdisziplinären Entwurfsprozess die Verwirklichung von 3D-Benutzerschnittstellen zu vollziehen, die zu einer quantitativen und qualitativen Weiterentwicklung von 3D-Interfaces führen wird.

### **Beitrag für das Forschungsfeld der Technischen Visualistik**

Diese Arbeit fokussiert die bildgestützte Interaktion von Anwendern mit interaktiven 3D-Anwendungen. Wesentliche Resultate, die einen Beitrag zu diesem Forschungsfeld leisten, konnten in internationalen Veröffentlichungen vorgestellt werden. Die Publikationen thematisieren unter anderem multiprojektive Abbildungsverfahren in [WOJDZIAK U. A. 2011E], nichtlineare Darstellungen in [WOJDZIAK U. A. 2011B] sowie multiple Sichten in [WOJDZIAK U. A. 2011C]. Weiterhin wird die Arbeitsumgebung Bildsprache LiveLab im Entwurfskontext (vgl. [KAMMER U. A. 2012]) und im Anwendungskontext (vgl. [WOJDZIAK U. A. 2011A]) vorgestellt. Die folgende Aufzählung fasst die wissenschaftlichen Hauptbeiträge dieser Arbeit zusammen.

- Die Identifikation von Modifikationsmöglichkeiten im computergrafischen Projektionsschritt zur Erzeugung linearer, nichtlinearer und multiprojektiver Darstellungsformen
- Die Systematisierung bildstruktureller Eigenschaften computergrafischer Darstellungen auf Basis eines modifizierbaren Kameramodells als Beitrag zur Weiterentwicklung und Verbreitung von nicht-fotorealistischen 3D-Darstellungen
- Die Formulierung von Gestaltungshinweisen für den 3D-Interfaceentwurf zur Entwicklung eines Grundverständnisses für den strukturellen Aufbau und die bildliche Ausgestaltung von 3D-Benutzerschnittstellen
- Die Klassifikation von Abbildungsverfahren hinsichtlich aufgabenorientierter Visualisierungsziele, womit eine Aufstellung an Verfahren für den anwendungsbezogenen Einsatz in interaktiven 3D-Anwendungen bereitgestellt ist
- Die Verfügbarkeit einer interaktiven 3D-Anwendung als serviceorientiertes und komponentenbasiertes System zur Integration von konfigurierbaren Visualisierungsbausteinen, wodurch Abbildungsverfahren für den 3D-Interfaceentwurf in einer interaktiven Echtzeitumgebung verfügbar sind



- Die Einführung eines neuartigen Vorgehens für einen schrittweisen und geführten Entwurfsprozess zur Realisierung interaktiver 3D-Benutzerschnittstellen, auf dessen Basis eine effiziente Gestaltung und Entwicklung erfolgen kann
- Die Entwurfsmuster als schablonenhafte Lösungsvorlagen für wiederkehrende Herausforderungen im Umgang mit nichtlinearen Darstellungsweisen und multiplen Sichten im Entwurfsprozess interaktiver 3D-Benutzerschnittstellen
- Die Entwicklung einer deklarativen Auszeichnungssprache zur Beschreibung eines Dialogbildes für ein bestehendes Datenbild auf Basis einer XML-basierten Grammatik für die einheitliche Beschreibung von Interaktionsbildern für 3D-Anwendungen

## Eingrenzung der Ergebnisse

Nachfolgend wird stichpunktartig eine Eingrenzung der Ergebnisse dieser Arbeit vorgenommen. Das umfasst einerseits Abgrenzungen, die aus dem Ansatz und der gewählten Vorgehensweise resultieren, andererseits Einschränkungen, die aufgrund des zeitlichen Rahmens dieser Arbeit bestehen und infolgedessen in künftigen Arbeiten thematisiert werden sollten.

- Die Verwirklichung von nichtlinearen Darstellungsformen wird in der Arbeit unter Einbeziehung der computergrafischen Kamera erreicht. Nichtlineare Bildräume können alternativ auf Basis von Schattierern erzeugt werden. Der Einsatz von Schattierern eröffnet eine Wahlmöglichkeit bei der Bildraumerzeugung, der nicht in der Arbeit, jedoch in weiterführenden Betrachtungen thematisiert werden sollte.
- Durch den Fokus auf die Entwicklung einer konzeptionellen Grundlage zur Unterstützung des Entwurfsprozesses von 3D-Benutzerschnittstellen auf planaren Ausgabegeräten erfolgte keine Einbindung umschließender und nicht-planarer Präsentationsmedien sowie deren Freiheitsgrade in die Betrachtungen der Arbeit.
- Der interaktive Dialog zwischen dem Nutzer und einem Anwendungssystem, mithilfe neuartiger Interaktionstechniken und Tracking-Technologien, war kein thematischer Bestandteil der Arbeit. Die methodische Aufarbeitung und die konkrete Ausgestaltung impliziter und expliziter Nutzerinteraktion unter Einsatz der Technologien ist ein zusätzlicher Aspekt im Nutzer-Bild-Dialog.
- Die Ausführungen in Bezug auf die 3D-Interfaceentwicklung erfolgen in der Arbeit auf Basis eines gleichbleibenden Objektraums. Die Analyse, Aufarbeitung und Methodenentwicklung für den Entwurf von Benutzerschnittstellen, unter Berücksichtigung raumzeitlicher Daten und dem Einsatz der „Zeit“ als Mittel der Gestaltung, werden nicht thematisiert.
- Die anwendungsunabhängige Verfügbarkeit und Übertragbarkeit eines Dialogbildes durch die abstrakte Beschreibung auf Basis einer Auszeichnungssprache macht Formattransformationen notwendig. Der Übersetzungsprozess in verschiedene 3D-Beschreibungsformate mit unterschiedlicher Mächtigkeit wird in der Arbeit nicht ausgeführt.

- Die Nutzung von BiLL als Arbeits- und Experimentierumgebung zur Erforschung und Entwicklung interaktiver 3D-Benutzerschnittstellen wurde anhand von Beispielszenarien aufgezeigt. Der Einsatz der Bildsprache LiveLab in realen Anwendungskontexten unter Einbindung tatsächlicher Datenbestände wurde in der Arbeit nicht ausgeführt.
- Der geführte 3D-Interfaceentwurf unter Einbindung der Entwurfsmuster kann auf Basis des Mosaik-Vorgehens vollzogen werden. Die Konzeption und Realisierung von Softwarewerkzeugen zur Unterstützung des Entwurfsprozesses, aufgrund des bestehenden Formalisierungs- und Abstraktionsgrades, wird in dieser Arbeit nicht thematisiert.
- Der Entwurfsprozess von 3D-Benutzerschnittstellen wurde in der Arbeit aufgezeigt, infolgedessen Interfaces für Anwendungssysteme realisiert werden können. In dieser Arbeit erfolgen jedoch keine Nutzerstudien, da derzeit keine Verfahren für eine objektive Qualitätsbestimmung der Nutzungsformen nicht-fotorealistischer Darstellungstechniken existieren.

## 7.3 Zukünftige Arbeiten

Im Folgenden wird ein Ausblick auf weiterführende Forschungsarbeiten gegeben. Die dargelegten Forschungsfragen und Herausforderungen bauen direkt auf den Ausführungen und den Erkenntnissen dieser Arbeit auf. Die Beschreibungen bieten eine perspektivische Sicht auf das Forschungsfeld der Technischen Visualistik und sollen gleichzeitig eine Motivation für weitere Forschungsarbeiten in diesem Themenfeld sein.

### Erweiterung des modellbasierten Vorgehens auf den Bildsyntheseprozess

Die Erläuterungen zur computergrafischen Bildsynthese im dritten Kapitel dieser Arbeit veranschaulichen die Verfahrensschritte, die zur Erzeugung eines computergrafischen Bildes notwendig sind. Die detaillierte Analyse des Projektionsschrittes zeigte verschiedene Modifikationsmöglichkeiten, die zur Veränderung der bildstrukturellen Eigenschaften genutzt und in der Folge für die bildgestützte Interaktion eingesetzt werden können. Für eine zukünftige Erweiterung der Vielfalt an Darstellungsweisen ist die Entwicklung zusätzlicher Abbildungsverfahren dienlich. Hierbei sollten die weiteren Schritte der Rendering-Pipeline einbezogen werden, da der Geometrieschritt und ebenso der Rasterisierungsschritt verschiedene Manipulationsmöglichkeiten für die Interfacegestaltung bieten. Mit einer Modifikation der Modell- und Kameratransformation innerhalb der Bildsynthese können beispielsweise betrachterbezogene Objektausrichtungen oder die Ordnung der Elemente hinsichtlich deren Relevanz in einem konkreten Anwendungskontext erreicht werden. Weiterhin können im Schritt des Vertex-Shading durch eine nutzer- und kontextadaptierte Texturierung und Beleuchtung von Objektraumelementen beispielsweise Veränderungen in Bezug auf deren Bedeutung, Relevanz und Akzentuierung realisiert werden. Ferner ist der Einsatz von Farben in der Bildgestaltung durch Fragment Schattierer innerhalb des Rasterisierungsschrittes umsetzbar und wird in [KAMMER U. A. 2013] im Kontext interaktiver Anwendungen sowie in [GOOCH U. A.

1998] und in [FRANKE & GROH 2006] bereits im Hinblick auf die Gestaltungsmöglichkeiten näher betrachtet.

Zur Gestaltung von Benutzerschnittstellen interaktiver 3D-Anwendungen sind in zukünftigen Arbeiten nach einer initialen bildgestalterischen Analyse von Darstellungstechniken diese algorithmisch zu beschreiben und als Softwarekomponenten für interaktive 3D-Anwendungen zur Verfügung zu stellen. Es ist weiterhin die Einbindung der Visualisierungstechniken in das modellbasierte Vorgehen zu vollziehen und infolgedessen die bestehenden UML-Profile zu erweitern beziehungsweise durch zusätzliche UML-Profile einer bereicherten Gestaltungsvielfalt anzupassen. Darüber hinaus sollten durch die Entwicklung von Entwurfsmustern die gewonnenen Erkenntnisse in Gestaltungsvorlagen überführt werden. Gleichfalls ist die XML-basierte Beschreibung des Dialogbildes durch eine definierte Knotenmenge zu erweitern, um den Sprachumfang entsprechend der Gestaltungsmöglichkeiten ebenfalls anzupassen.

Jeder dieser Arbeitsschritte ist aufbauend auf den Ergebnissen der Arbeit notwendig, da bisher kein standardisiertes Vorgehen für das Visualisierungsdesign von 3D-Benutzerschnittstelle existiert. Ferner bedingt die Analyse von Visualisierungstechniken sowie die Verallgemeinerung von Auswahl- und Gestaltungsregeln zusätzliche Forschungsarbeiten, um eine zukünftige Vereinheitlichung des Entwurfsprozesses von 3D-Benutzerschnittstellen zu erreichen.

### **Automatisierungsgrad des Nutzer-Bild-Dialoges**

Mit der Arbeitsumgebung Bildsprache LiveLab wurde eine interaktive 3D-Anwendung verwirklicht, die eine vornehmlich nutzerseitige Kontrolle des Visualisierungsprozesses bietet. Dies ist dem experimentellen Charakter der Anwendung geschuldet und einer Arbeitsumgebung im Forschungskontext angemessen. In nachfolgenden Arbeiten sollten jedoch die Rahmenbedingungen und Einflussfaktoren, die den Einsatz einer Darstellungstechnik und deren spezifische Charakteristiken sowie Parameter im Anwendungskontext bedingen, analysiert werden. Für die Verwirklichung einer expressiven, effektiven und angemessenen Visualisierung (vgl. [SCHUMANN & MÜLLER 2000, S. 12]) ist bisher das bestehende Expertenwissen (vgl. [TORY U. A. 2004] und [WARE 2004]), des Designers und des Entwicklers situativ anzuwenden.

Um im Visualisierungsdesign die Qualität eines Bildraums und ebenso die Effektivität des Nutzer-Bild-Dialogs zu steigern, ist die Identifikation der Parameter notwendig, die zur Festlegung eines zielgerichteten Darstellungsverfahrens notwendig sind. In zukünftigen Arbeiten sollten daher die Parameter linearer und nichtlinearer Darstellungstechniken der Ausgangspunkt für eine Eignungsbewertung hinsichtlich eines Einsatzes im Anwendungskontext sein. Dieser Schritt ist für ein allgemeingültiges Visualisierungsdesign von 3D-Benutzerschnittstellen oder eine Standardisierung von Darstellungsverfahren, Gestaltungsregeln und Beschreibungsformen notwendig. Durch die Analyse der Parameter von Visualisierungsverfahren, in Bezug auf die resultierenden bildstrukturellen Eigenschaften, kann eine teilautomatische Festlegung situations- und kontextabhängiger Darstellungstechniken abgeleitet und folglich vorkonfigurierte Verfahren in die Benutzerschnittstellen interaktiver

3D-Anwendungen integriert werden. Beispielsweise kann der Öffnungswinkel einer computergrafischen Kamera als Indikator für eine automatische Anwendung einer objektraumbasierten Multiperspektive zur Minderung perspektivischer Verzerrungen dienen (vgl. [YANKOVA & FRANKE 2008]). Ein besonderes Augenmerk ist in diesem Zusammenhang auf den interaktiven Prozess zwischen Nutzer und System zu richten. Mit dem Ziel den Nutzer-Bild-Dialog zu optimieren, ist der Grad der Automatisierung beziehungsweise der Interaktivität zu reflektieren. Einerseits ist es notwendig den Anwender in der bildgestützten Interaktion zu leiten, wobei dieser den Prozess gleichzeitig steuern sollte. Dabei ist zu beachten, dass mit einem zunehmenden Automatisierungsgrad die Akzeptanz einer 3D-Benutzerschnittstelle sinkt, wenn der Nutzer sich beim Erreichen von Visualisierungszielen übergangen fühlt. Andererseits muss bei einer geringen Entscheidungsunterstützung ein Problembewusstsein beim Nutzer erzeugt werden, damit dieser eine geeignete Lösung selbstständig finden kann.

### **Implizite Nutzerinteraktion für adaptive 3D-Benutzerschnittstellen**

Die Betrachtungen in dieser Arbeit fokussieren die Entwicklung eines methodischen Ansatzes, um den Entwurfsprozess von 3D-Benutzerschnittstellen im Hinblick auf einen nutzerzentrierten und gebrauchstauglichen Nutzer-Bild-Dialog zu unterstützen. Für die Etablierung von Benutzerschnittstellen unter Berücksichtigung einer wahrnehmungskonformen und adaptiven Kommunikation mit dem Nutzer sind die Ergebnisse der Arbeit ein wesentlicher Schritt. Hierbei wurde der Anwender als Rezipient in die Interfaceerzeugung einbezogen, der durch explizite Aktionen eine Benutzerschnittstelle in einem interaktiven Prozess anpassen kann. Darüber hinaus kann der Nutzer-Bild-Dialog durch implizite Interaktionen erweitert werden, bei denen der Anwender Aktionen ausführt, die nicht primär der Kommunikation mit einem interaktiven System dienen, diese aber von der Anwendung als Eingabe verstanden wird. Damit eine Benutzerschnittstelle auf implizite Aktionen des Nutzers reagieren kann, muss das System die Aktionen beziehungsweise Zeichen des Nutzers erkennen und interpretieren können (vgl. [SCHMIDT 2000, S. 192]). Wesentliche Faktoren sind in diesem Zusammenhang die Position der Person, deren Blickrichtung und Bewegung sowie deren persönliche Charakteristiken (vgl. [BALLENDAT U. A. 2010, S. 124; STREITZ U. A. 2003, S. 134–135]). Die gewonnenen Nutzerinformationen müssen durch das System im Anwendungskontext interpretiert werden und eine entsprechende Anpassung des computergrafischen 3D-Interfaces auslösen. Die Erweiterung der Gestaltungsmöglichkeiten durch implizite Nutzerinteraktionen sollte durch die weiterführende Erforschung von Tracking-Technologien und deren Integration in interaktive 3D-Anwendungen forciert werden. In der Folge kann durch die Identifikation des Nutzers und die Positionsbestimmung vor dem Interface der Bildraum implizit auf den Anwender angepasst werden und ein individueller Nutzer-Bild-Dialog etabliert werden, der beispielsweise durch die Anpassung der geometrischen Mitte an die Kopfposition des Nutzers oder durch die Adaption der dargestellten Informationen an das Alter eines Nutzers verbildlicht wird (vgl. [WOJDZIAK U. A. 2011D]). Weiterhin ist die Entwicklung von Methoden für die Anpassung von 3D-Benutzerschnittstellen an das Blickverhalten des Anwenders notwendig, um dessen Aufmerksamkeit zu identifizieren und die Visualisierung nutzerunterstützend anzupassen (vgl. [KUSCH U. A. 2010]). Ein Anwender kann infolgedessen bei der Exploration von Daten oder bei der Suche nach Informationen durch eine implizite bildgestützte Interaktion unterstützt werden.

## Durchführung von Nutzerstudien

Ein wichtiges Vorgehen zur Beurteilung interaktiver 3D-Benutzerschnittstellen ist die Durchführung von Studien. Die Überprüfung der Gebrauchstauglichkeit einer Schnittstelle auf Basis von Anwendungsfällen gewinnen hierbei aufgrund ihres Potenzials für die Mensch-Maschine-Kommunikation und in Ermangelung gleichwertiger Alternativen zunehmend an Bedeutung (vgl. [KOSARA U. A. 2003; TORY & MÖLLER 2004]). Der derzeitige Forschungsstand von generalisierten computergrafischen Darstellungen bietet keine objektiven Maße, formalen Modelle oder gesicherten Berechnungsverfahren für eine Qualitätsbestimmung grafischer Darstellungen in interaktiven 3D-Systemen. In der Folge ist die Identifikation und Beschreibung von Anwendungsszenarien durch heterogene Benutzergruppen ein initialer Schritt und gleichzeitig eine notwendige Voraussetzung für die Kumulation von Informationen und Erfahrungen im Hinblick auf die konkrete Ausgestaltung von 3D-Benutzerschnittstellen. Aufbauend auf den konzeptionellen und methodischen Ergebnissen der Arbeit sollten in nachfolgenden Schritten Nutzerstudien zur Evaluation modellbasierter 3D-Benutzerschnittstellenentwürfe durchgeführt werden. Der Konzeption der Nutzerstudien sollte zum einen ausgehend von den in der Arbeit gegebenen Gestaltungshinweisen erfolgen, zum anderen das existierende Expertenwissen aus dem Bereich der Interface und Interaktionsgestaltung einbeziehen, das unter Fokussierung verschiedener Aspekte der Mensch-Computer-Interaktion veröffentlicht wurde. Diesbezüglich sind unter anderem die Arbeiten zum Thema „*Human Factors in Visualization Research*“ (vgl. [TORY & MÖLLER 2004]) oder die Ausführungen zu wahrnehmungsbasierten Entwurfsrichtlinien (vgl. [WARE 2004]) zu nennen. Für eine Evaluation von 3D-Interfaces ist die Überprüfung einzelner bildstruktureller Aspekte und deren Zusammenhänge in einer Benutzerschnittstelle notwendig, wodurch Entscheidungsunterstützungen für den Einsatz von Abbildungsverfahren sowie Kompositions- und Montageschritte abgeleitet werden können. Beispielsweise bietet die Identifikation eines konkreten Kameraöffnungswinkels, wie dieser in [YANKOVA & FRANKE 2008] als Schwellwert identifiziert wurde, eine Entwurfsunterstützung für den Einsatz multiprojektiver Abbildungstechniken zur Minderung perspektivischer Verzerrungen in computergrafischen Darstellungen.

Auf Basis der Ergebnisse von Nutzerstudien kann langfristig die Gebrauchstauglichkeit von 3D-Interfaces, unter Berücksichtigung der Anforderungen, der Daten und des Nutzers, verbessert werden, indem interaktive 3D-Anwendungen an die physischen und mentalen Fähigkeiten von Menschen angepasst werden.

## Weiterentwicklung des Bildsprache LiveLab

Das Bildsprache LiveLab ist einer ständigen Wartung, Optimierung und Erweiterung unterworfen. Die Entwicklungen erfolgen im Hinblick auf Performanzverbesserungen und auf die Anpassung von Schnittstellen, um die Einbindung von Visualisierungsbausteinen zu vereinfachen. Weiterhin erfolgen die fortwährende Konzeption und Realisierung von Abbildungsverfahren und deren methodische Integration in den Entwurfsprozess von 3D-Benutzerschnittstellen sowie die softwaretechnologische Umsetzung in der Arbeitsumgebung. Nachfolgend werden wesentliche Schwerpunkte zukünftiger funktionaler Entwicklung stichpunktartig aufgeführt und erläutert. Diese thematisieren einerseits die Erfüllung bisher nicht umgesetzter

Anforderungen (siehe Abschnitt 4.1), andererseits werden Aspekte zur weiterführenden Unterstützung der Forschungstätigkeit im Bereich der Technischen Visualistik benannt:

- Die Verfügbarkeit von Benutzeroberflächen im Editor-Fenster, die für unterschiedliche Anwender- und Personenkreise entworfen und umgesetzt sind. Dadurch ist der Einsatz der Arbeitsumgebung auch in nicht-technischen Bereichen gewährleistet.
- Die Bereitstellung der BiLL-Anwendung auf den Betriebssystemen OS X und Linux, um die Verfügbarkeit (im wissenschaftlichen Umfeld) zu erhöhen.
- Die Konzeption und Entwicklung von „Bundle-Familien“ auf Basis einer weiterentwickelten Servicearchitektur, um spezifische Anwendungskonfigurationen, beispielsweise für den Bereich der Produktentwicklung oder der Systembiologie, bereitstellen zu können. In der Folge ist eine adaptierte Ausprägung der Arbeitsumgebung für spezifische Nutzergruppen umsetzbar.
- Die Erweiterung und Anpassung von Visualisierungs- und Interaktionsverfahren ist notwendig, um deren Funktionsweise auf nichtplanare und umschließende Ausgabegeräte anzupassen sowie unter Einbindung neuartige Eingabetechnologien einsetzen zu können.
- Die Erweiterung der Graphendarstellung im Editor-Fenster zu einem vollwertigen und interaktiven Interaktionsbildgraphen. Durch die Repräsentation des Daten- und des Dialogbildes mit deren Eigenschaften und Parametern sowie unter Berücksichtigung von Parameteränderungen im zeitlichen Verlauf kann der interaktive Prozess des Nutzers mit der Anwendung auf Grundlage einer graphbasierten Abstraktion des Interaktionsbildes unterstützt werden.
- Die Erweiterung des BiLL-Editors um eine abstrakte, schematische Repräsentation des strukturellen Aufbaus der Bildsynthese; inklusive der Eigenschaften und Parameter sowie der vollzogenen Modifikationen. Dadurch wird das Verständnis für die Vorgänge bei der Manipulation der computergrafischen Bildsynthese verbessert, da die erzeugten Bildräume zusätzlich in einer schematischen Form beschrieben und visualisiert werden.

## Werkzeugunterstützung für den Entwurfsprozess

Die Verwirklichung einer gegebenen Anforderungsspezifikation in einem effektiven und effizienten Entwurfsprozess zur Verwirklichung einer 3D-Benutzerschnittstelle ist heute bereits in vielen Bereichen ein entscheidender Faktor im Hinblick auf die Umsetzbarkeit und spätere Akzeptanz einer interaktiven 3D-Anwendung. In der Arbeit wird mit dem Bildsprache LiveLab bereits ein Werkzeug vorgestellt, das die Anwendung des modellbasierten Vorgehens in der Phase des finalen Interfaceentwurfs unterstützt. In zukünftigen Arbeiten ist es jedoch notwendig, das Bildsprache LiveLab um Funktionen zur werkzeuggestützten Unterstützung sämtlicher Phasen des Mosalk-Vorgehens zu erweitern und diese methodisch in einem ganzheitlichen Entwurfsprozess zu fusionieren. Ein solches Softwarewerkzeug muss durch die Einbindung sämtlicher Vorgehensschritte des Mosalk-Verfahrens und der zugrunde liegenden Methoden sowie deren enge Verzahnung gekennzeichnet sein. In der Folge sollte ein

vollständig werkzeuggestützter Entwurfsprozess realisiert werden, der den bidirektionalen Entwicklungsverlauf zwischen initialen Anforderungen an ein 3D-Interface und einer implementierten Benutzerschnittstelle unterstützt und in der Folge eine Transparenz und Nachvollziehbarkeit im Designprozess sicherstellt.

Ausgehend von einer Zeichenfläche als Benutzerschnittstelle könnte der Entwurfsprozess in einem Softwarewerkzeug vollzogen werden, wobei ein Zoom-Konzept für die Verknüpfung der Phasen (als Zoom-Ebenen) innerhalb des modellbasierten Vorgehens eingesetzt wird (vgl. [MEMMEL 2009, S. 184 ff.]). Ausgehend von identifizierten Anwendungsszenarien kann ein semantischer Zoom zur Veränderung der Darstellung und des Granularitätsgrades eines Interfaceentwurfes in Abhängigkeit der Phasen des MosalK-Vorgehens eingesetzt werden. Das Zoomable User Interface<sup>65</sup> (ZUI) ermöglicht zwischen den unterschiedlichen Abstraktionsebenen zu wechseln und unter Beibehaltung eines visuellen Zusammenhangs die modellierten Entwürfe und die finale Interfacebeschreibung in Relation zueinander zu setzen. Dadurch können die Erstellung eines Interface einerseits und die Überprüfung von Nutzungsanforderungen andererseits in einem gesamtheitlich werkzeuggestützten Prozess erfolgen. In der Phase des konkreten 3D-Interfacedesigns kann darüber hinaus zur modellhaften Skizzierung des Interface die tatsächliche Ausgestaltung als gerenderte Benutzerschnittstelle in das Werkzeug integriert sein, um den Entwurfsprozess für Designer und Entwickler visuell zu unterstützen. Die Gestaltung des Verhaltens einer Benutzerschnittstelle kann ferner durch ein Echtzeit-Rendering begünstigt werden, wie es durch die Bildsprache LiveLab bereitgestellt wird, um bildstrukturelle Veränderungen im Interaktionsprozess direkt zu kommunizieren.

## 7.4 Abschließende Bemerkung

Die bestehenden Einflussfaktoren – beispielsweise der Anwendungskontext in dem eine interaktive 3D-Anwendung eingesetzt werden soll, die Nutzeraufgaben, deren Erfüllung durch eine interaktive Anwendung zu erfolgen hat sowie die mit einem System verknüpften Daten – sind der Ausgangspunkt für den Entwurfsprozess einer Benutzerschnittstelle. Durch die Verfügbarkeit von analysierten und bewerteten Abbildungsverfahren, aus denen einerseits Gestaltungshinweise abgeleitet und andererseits abstrakte Entwurfsvorgaben herausgearbeitet werden konnten, beschreibt diese Arbeit einen Unterstützungsprozess für die Entwicklung interaktiver 3D-Benutzerschnittstellen. Im Rahmen dieser Arbeit konnten durch Regeln, Methoden und Vorlagen die Grundlagen für den effektiven und effizienten Einsatz von nicht-fotorealistischen Darstellungsformen in 3D-Benutzerschnittstellen interaktiver Systeme gelegt werden. Sicherlich kann darauf aufbauend in den nächsten Jahren die Vielfalt, Attraktivität und vor allem die Effektivität von 3D-Benutzerschnittstellen gesteigert und die bildgestützte Interaktion als ein nutzerabhängiger Dialog gestaltet werden.

<sup>65</sup> *Zoomable User Interface* engl. Skalierbare Benutzerschnittstelle; Grafische Benutzerschnittstelle, welche die Möglichkeit bietet, die Elemente eines Interface oder die gesamte Benutzerschnittstelle in unterschiedlichen Skalierungsgraden darzustellen.



# Anhang

## Legende zu den UML-Profilen und den Entwurfsmustern

Die Entwurfsmuster, die in dieser Arbeit vorgestellt werden, basieren auf einer einheitlichen Syntax, welche an die UML-Notation von Klassendiagrammen der Softwaretechnologie angelehnt sind und durch die UML-Profile Mosalk:Abstrakt und Mosalk:Konkret an den Anwendungskontext des 3D-Interfaceentwurfes angepasst sind.

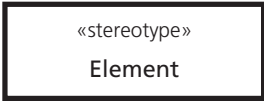
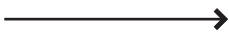



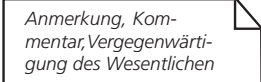
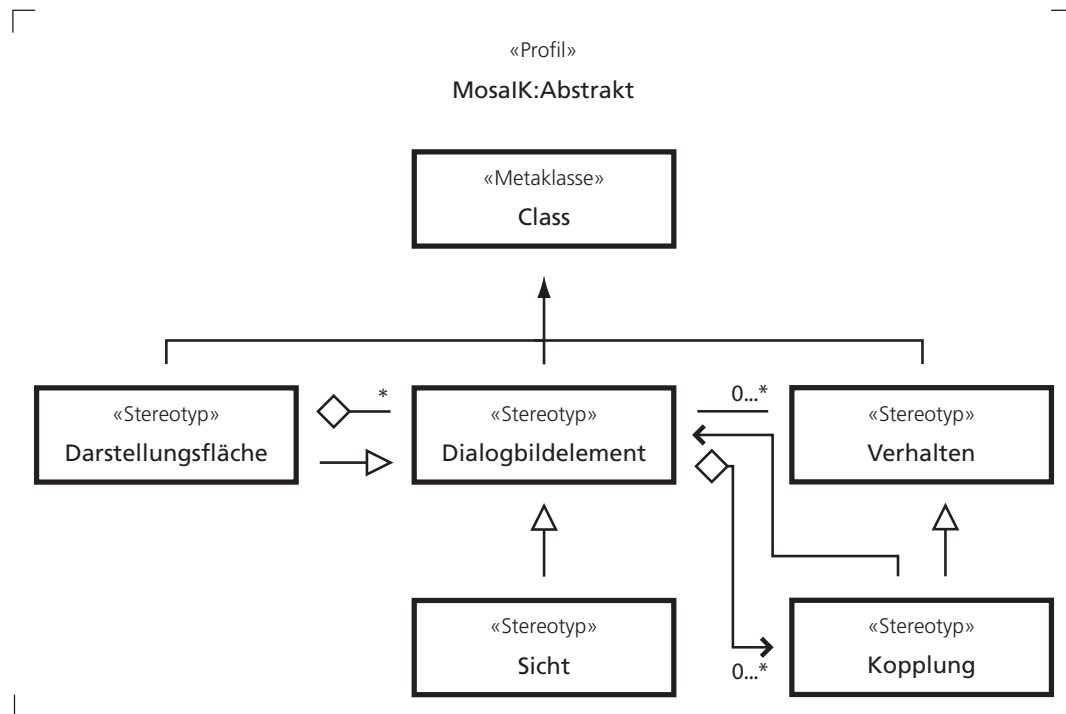
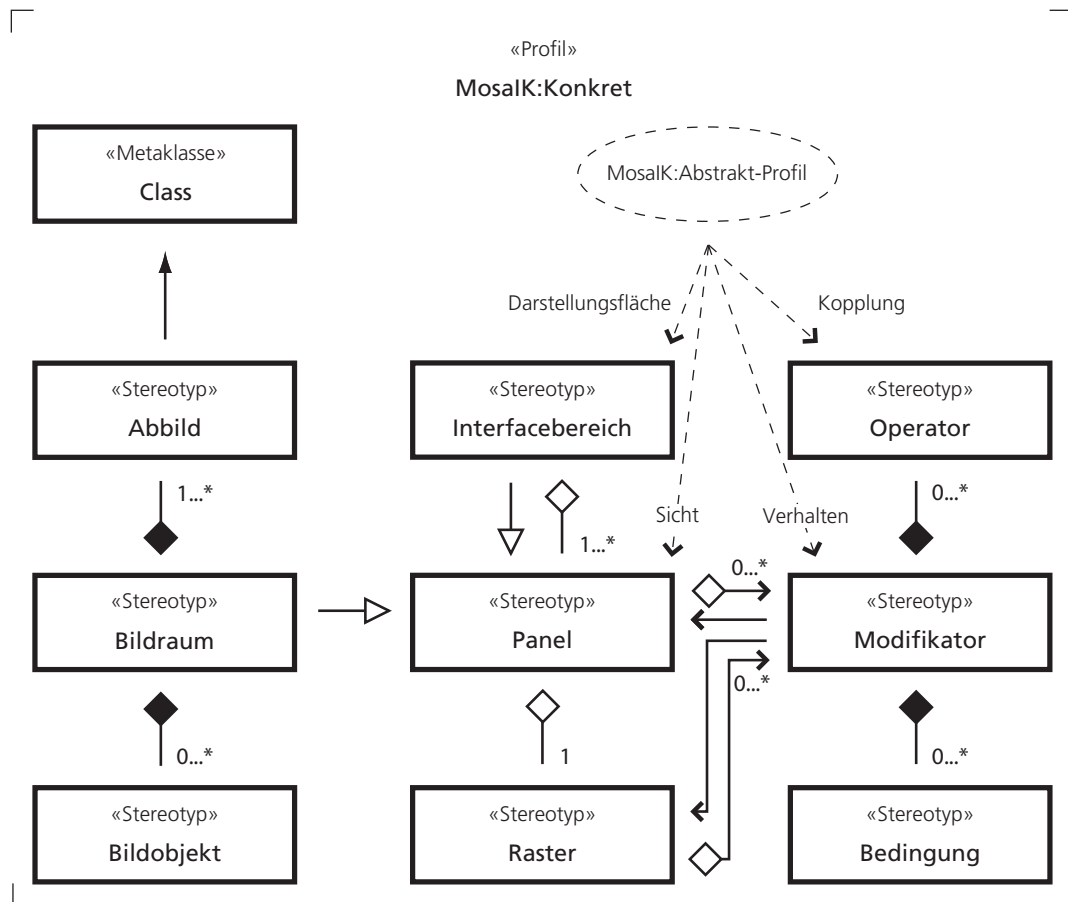
Abbildung	Erläuterung
	Ein Element repräsentiert einen Bestandteil (Aspekt) einer 3D-Benutzerschnittstelle im Entwurfsprozess innerhalb der Modellierungsphase. Das Element kann einerseits ein visueller Teil der Benutzerschnittstelle sein (beispielsweise eine Sicht), andererseits ein Verhalten des 3D-Interface (beispielsweise eine Sichtenkopplung) beschreiben
	Eine Assoziation ist eine n-stellige Relation zwischen zwei Elementen, die zueinander in (gerichteter oder) unspezifischer Beziehung stehen.
	Die Vererbung skizziert eine Beziehung zwischen zwei Elementen, bei der das Oberelement (Elternelement) eine Generalisierung des Unterelementes (Kindelement) beschreibt.
	Eine Aggregation ist eine Beziehung zwischen Elementen einer Benutzerschnittstelle, bei der ein Element einer Menge weiteren Elementen übergeordnet ist.
	Eine Komposition ist eine spezielle Beziehung zwischen Elementen einer Benutzerschnittstelle, bei der ein Element weiteren Elementen übergeordnet ist. In dieser Relation kann das untergeordnete Element nicht ohne das übergeordnete Element existieren.
	Das Kommentarfeld ist optional und ermöglicht die Integration von weiterführenden Anmerkungen und Hinweisen in die Modelle der Benutzerschnittstellen. Ein Kommentar kann über eine Verbindung einem Element zugeordnet sein oder als beschreibende Ergänzung einen Bezug zum gesamten Interfacemodell aufweisen.

Tabelle 14: Legende zu den UML-Profilen



Stereotyp	Erläuterung
Darstellungsfläche	Die Darstellungsfläche beschreibt einen Gruppierungscontainer, der als Grundfläche einer Benutzerschnittstelle dient und auf der Interfaceelemente platziert werden können.
Dialogbildelement	Das Stereotyp beschreibt alle Interfacebestandteile in einer generellen Form und dient der Abstraktion der verschiedenen Benutzerschnittstellenelemente.
Sicht	Hierdurch wird eine gerahmte lineare oder nichtlineare Darstellung eines zugrunde liegenden Objektraums in einer 3D-Benutzerschnittstelle beschrieben.
Verhalten	Interfaceelemente können ein Verhalten aufweisen, weshalb deren Veränderlichkeit durch das Stereotyp beschrieben werden kann.
Kopplung	Das Stereotyp Kopplung ist ein Spezialfall eines Verhaltens, mit dem Veränderungen voneinander abhängigen Dialogbildelementen beschrieben werden können.

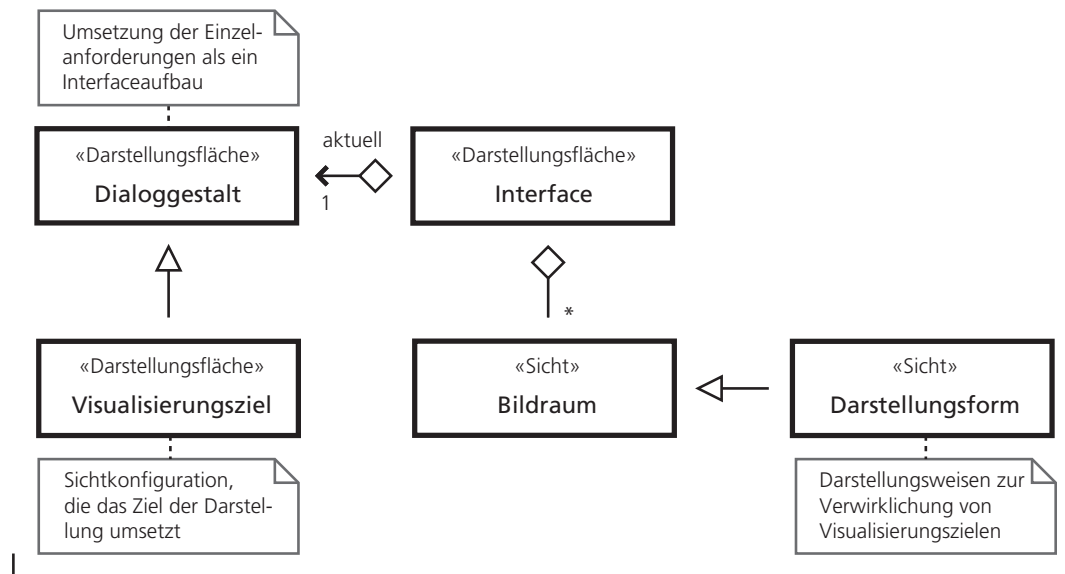
Tabelle 15: Übersicht über das MosaIK:Abstrakt-Profil



Stereotyp	Erläuterung
Interfacebereich	Der Interfacebereich ist eine Konkretisierung des Stereotyps Darstellungsfäche des MosaIK:Abstrakt-Profils und repräsentiert die Grundfläche einer 3D-Benutzerschnittstelle.
Panel	Das Panel konkretisiert das Sichtelement des MosaIK:Abstrakt-Profils und beschreibt eine gerahmte Abbildfläche, durch die eine Abgrenzung zwischen Visualisierungen erfolgt.
Raster	Das Element fungiert als Grundlinienraster für die Anordnung von Panels. Hierdurch kann einerseits die Grundfläche einer Benutzerschnittstelle zur Anordnung von Panels gerastert werden, andererseits wird jedes Panel zur Komposition von Bildräumen mit einem Raster versehen.
Bildraum	Der Bildraum ist ein Spezialfall eines Panel-Elements und repräsentiert eine lineare, nichtlineare oder multiprojektive Darstellung eines Objektraums.
Abbild	Dieses Stereotyp repräsentiert das Resultat eines Projektionsvorganges auf Basis eines linearen Abbildungsverfahrens, das einen monoperspektivischen Bildraum erzeugt oder einen Teilbild eines multiperspektivischen Bildraums bildet.
Bildobjekt	Mit diesem Stereotyp werden Elemente des Objektraums repräsentiert.
Modifikator	Der Modifikator ist eine Konkretisierung des Stereotyps Verhalten des MosaIK:Abstrakt-Profils und repräsentiert ein Gruppierungscontainer, der eine Anzahl von Einzeloperationen zur Veränderung einer 3D-Benutzerschnittstelle aufnehmen kann.
Operator	Das Stereotyp ist eine Konkretisierung des Stereotyps Verhalten des MosaIK:Abstrakt-Profils und repräsentiert eine diskrete oder eine kontinuierliche Veränderung einer Eigenschaft eines Panels oder eines Bildraums.
Bedingung	Das Stereotyp beschreibt Gegebenheiten, die für das Verhalten von 3D-Benutzerschnittstellen bestimmend sind und wodurch dessen Verhalten beeinflusst und gesteuert werden kann.

Tabelle 16: Übersicht über das MosaIK:Konkret-Profil

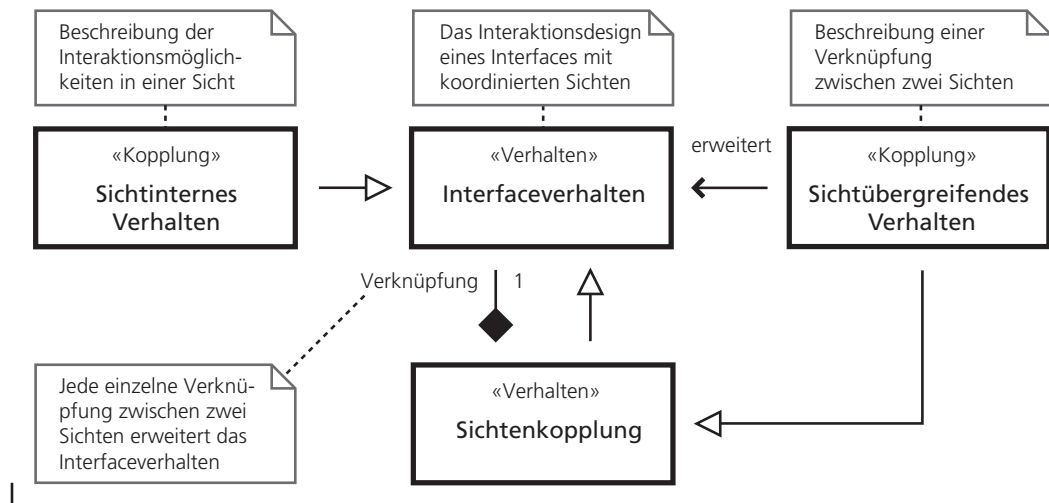
## Entwurfsmuster: Dialoggestalter



Merkmal	Erläuterung
Name	Dialoggestalter
Problemkontext	Im Nutzer-Bild-Dialog dienen veränderliche Darstellungen eines Objektraums dem interaktiven Prozess zwischen Anwender und System. In Abhängigkeit von Nutzungsanforderungen sowie dem Nutzer und dessen Aufgaben variieren die bildstrukturellen Charakteristiken der Bildräume. Durch den interaktiven Charakter eines 3D-Interface ist eine einmalige Festlegung der Anzahl und der Ausgestaltung von Sichten unzureichend.
Problem	Die Bestimmung der bildstrukturellen Eigenschaften von Bildräumen und die Festlegung der Sichtenanzahl für ein 3D-Interface sind nicht trivial. Vielmehr können der Aufbau sowie die Anzahl von Sichten durch einen inadäquaten Einsatz den Nutzer-Bild-Dialog stören und die Darbietung von Informationen negativ beeinflussen.
Ebene	Abstraktes Interfacedesign
Lösungsvorschlag	Der bildstrukturelle Aufbau einzelner Bildräume sowie deren Anzahl werden für die zu erfüllenden Visualisierungsziele festgelegt und als konkrete Sichten auf dem Darstellungsfeld für die Ausgestaltung einer Benutzerschnittstelle arrangiert.
Vor- und Nachteile	<ul style="list-style-type: none"> <li>+ Der Interfaceentwurf erfolgt mit direktem Bezug zu konkreten Visualisierungszielen.</li> <li>+ Das Zusammenwirken von Bildräumen kann bereits im Entwurf konzipiert und Inkonsistenzen in der Gestaltung verhindert werden.</li> <li>+ Eine Erweiterbarkeit und Flexibilität ist durch die Aufteilung einer Benutzerschnittstelle in einzelne Ausgestaltungen gewährleistet</li> <li>+ Die abstrakte Ausgestaltung des Aufbaus der Benutzerschnittstelle dient der frühzeitigen Validierung des Nutzer-Bild-Dialoges</li> <li>- Durch die Einbindung der Dialoggestalt als zusätzliche Abstraktionsebene führt der Gebrauch des Musters bei einfachen Interfaceentwürfen zu einem erhöhten Aufwand</li> <li>- Es werden keine inhärenten Hinweise für den zweckmäßigen Einsatz von Darstellungsformen geboten</li> </ul>
Verwendung	Interfaces, die verschiedene Visualisierungsziele unterstützen sollen, beispielsweise die Umsetzung des Visual Information Seeking Mantra. Das Prinzip findet in GeoWizard (vgl. [FELDT U. A. 2005]) und im Bildsprache LiveLab (vgl. [STARKE U. A. 2011]) eine Anwendung
Musterverbindungen	Es bestehen Verbindungen zum Regisseur auf der Ebene des abstrakten Interfacedesigns und zu den Mustern Raster und Sichtweise auf der Ebene des konkreten Interfacedesigns

Tabelle 17: Kurzdarstellung des Entwurfsmusters Dialoggestalter

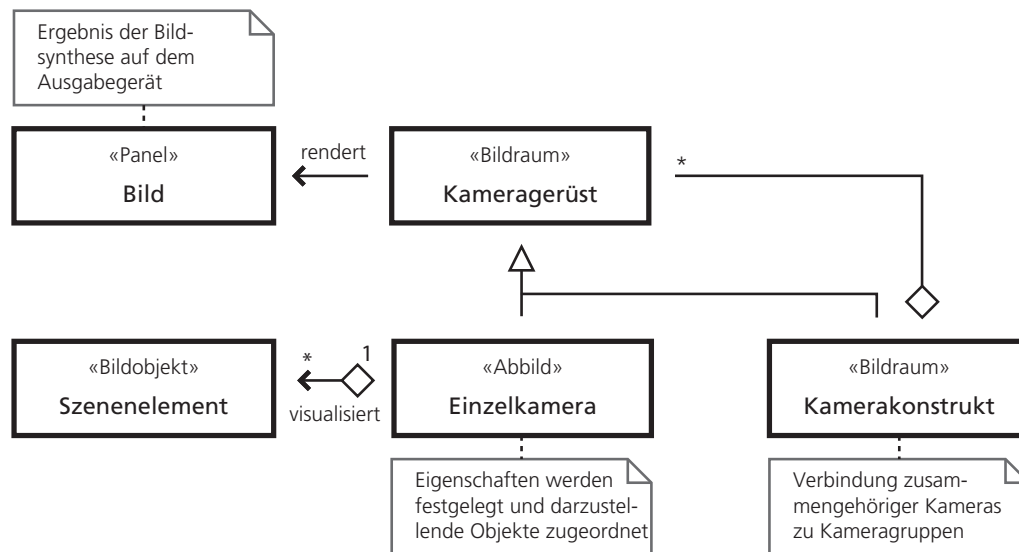
## Entwurfsmuster: Regisseur



Merkmal	Erläuterung
Name	Regisseur
Problemkontext	Der interaktive Prozess zwischen Nutzer und System bewirkt eine fortwährende Abwandlung des Erscheinungsbildes der 3D-Benutzerschnittstelle, indem sowohl Anpassungen an der Bildstruktur als auch am Verhalten erfolgen. Durch die verfügbaren Freiheitsgrade innerhalb einer 3D-Sicht einerseits und durch die Verknüpfungsvarianten von Sichten basierend auf den Sichteigenschaften andererseits, besteht eine Vielzahl von Interaktionsmöglichkeiten, die im Entwurfsprozess eingesetzt werden können.
Problem	Ein unstrukturiertes Vorgehen bei der Festlegung des Verhaltens einer 3D-Benutzerschnittstelle kann daher zu einem intransparenten und inkonsistenten Interaktionsdesign führen.
Ebene	Abstraktes Interfacedesign
Lösungsvorschlag	Um in einem Interfaceentwurf die Konsistenz des Interaktionsdesigns und die Nachvollziehbarkeit der Abhängigkeiten sicherzustellen, erfolgt die Konzeption ausgehend von der einzelnen Sicht. Ein definierter Interaktionsumfang innerhalb einer jeden Sicht kann um zusätzliche Sichtenkopplungen ergänzt werden.
Vor- und Nachteile	<ul style="list-style-type: none"> <li>+ Durch die Trennung von sichtinternem und sichtübergreifendem Verhalten werden in der Interaktionsstruktur die Fallunterscheidungen minimiert</li> <li>+ Eine Kapselung und individuelle Spezifikation des Verhaltens von Einzelsichten vereinfacht den Entwurf des Interaktionsdesigns.</li> <li>+ Vermeidung eines inkonsistenten Interaktionsdesigns, da jede Sichtenkopplung lediglich eine Abhängigkeit beschreibt und die Kopplungen rekursiv beschrieben sind.</li> <li>- Durch eine große Anzahl von gegenseitigen Kopplungen kann die Übersichtlichkeit des Interaktionsdesigns beeinträchtigen</li> <li>- Die Identifikation von kontextspezifischen Unzulänglichkeiten kann bei einer großen Anzahl Abhängigkeiten erschwert sein</li> </ul>
Verwendung	Interaktive Systeme, die koordinierte multiple Sichten im Entwurf von 3D Modellen einsetzen oder Präsentationswerkzeuge zur Verdeutlichung von Zusammenhängen komplexer Daten; beispielsweise Autodesk Maya als 3D-Modellierungswerkzeug (vgl. [AUTODESK]) und MeVisLab für die Datenvisualisierung (vgl. [MEVISLAB])
Verbindungen zu anderen Mustern	Das Entwurfsmuster weist Verbindungen zum Dialoggestalter (abstraktes Interfacedesign) auf und wird durch das Muster Modifikation im konkreten Interfacedesign verfeinert.

Tabelle 18: Kurzdarstellung des Entwurfsmusters Regisseur

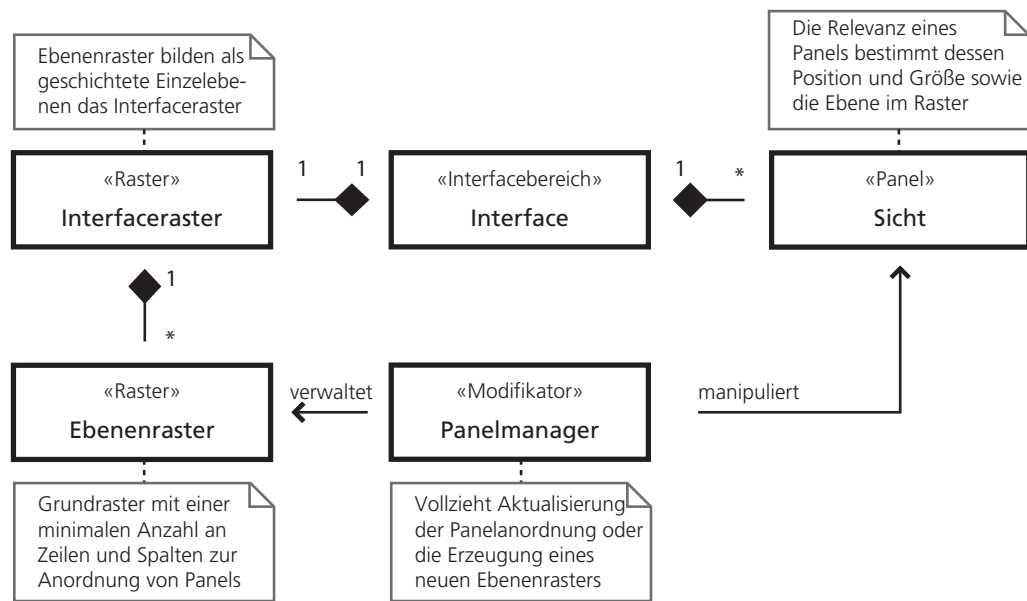
## Entwurfsmuster: Sichtweise



Merkmal	Erläuterung
Name	Sichtweise
Problemkontext	Für eine Reihe von Visualisierungszielen ist eine Beschränkung auf lineare Projektionen nicht opportun. Situativ erreichen krummlinige oder multiperspektivische Darstellungen einen zusätzlichen Nutzen für die Darstellung räumlicher Strukturen und sollten daher im Entwurf von 3D-Benutzerschnittstellen genutzt werden.
Problem	Der Entwurf computergrafischer Kameramodelle zur Erzeugung nichtlinearer und multiprojektiver Abbilder zur Verwirklichung eines Visualisierungsziels ist aufgrund der hohen Komplexität und der verfügbaren Freiheitsgrade der Kamera weder für den Designer noch den Entwickler eine triviale Aufgabe.
Ebene	Konkretes Interfacedesign
Lösungsvorschlag	Konstruktionen von einfachen sowie komplexen lineare, kurvilineare und multiprojektive Abbildungen mit individuellen Projektionseigenschaften, durch die Strukturierung von Kameras und Abhängigkeiten in einem computergrafischen Kamerakonstrukt auf Basis von Teil-Ganzes-Hierarchien
Vor- und Nachteile	<ul style="list-style-type: none"> <li>+ Die Festlegung von linearen, nichtlinearen und multiprojektiven Abbildungsverfahren ist über eine einheitliche Struktur möglich.</li> <li>+ Die Wiederverwendbarkeit und die Erweiterung von Kameragerüsten vereinfachen den Entwurfsprozess.</li> <li>+ Durch den modularen Ansatz können Kameramodelle erzeugt und während der Laufzeit geändert werden.</li> <li>- Die Festlegung, inwieweit Kameraoperationen innerhalb des Kameramodells oder im Kameragerüst definiert werden, ist nicht durch das Entwurfsmuster vorgegeben.</li> <li>- Es kann aufgrund einer vergleichsweise einfachen Integration einer großen Anzahl Kameras zu Performanceproblemen in der späteren Anwendung kommen.</li> </ul>
Verwendung	Für computergrafische Panoramadarstellungen, verzerrungsfreie Weitwinkelprojektionen und Fokus-und-Kontext-Darstellungen
Verbindungen zu anderen Mustern	Das Entwurfsmuster besitzt Verbindungen zum Modifikation-Entwurfsmuster für die Aktualisierung von Kameramodellen, zu Raster als Ordnungselement für Bildräume und zum Wandlermuster für eine situative Anpassung von Kameramodellen

Tabelle 19: Kurzdarstellung des Entwurfsmusters Sichtweise

## Entwurfsmuster: Panelraster

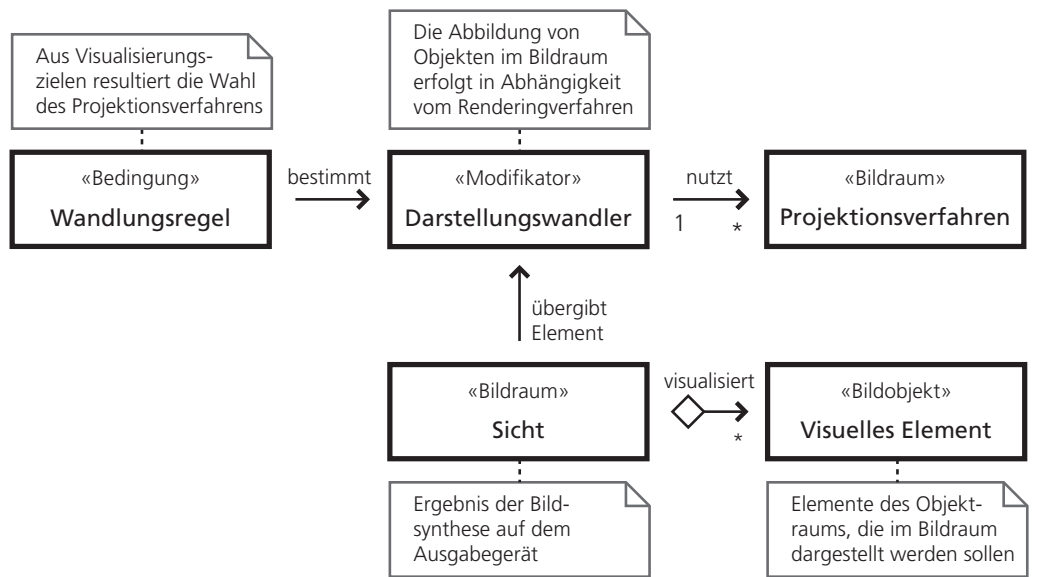


Merkmal	Erläuterung
Name	Panelraster
Problemkontext	Eine konkrete Nutzeraufgabe sowie ein gegebener Datenbestand können den Einsatz multipler Sichten bedingen. Der strukturelle Aufbau einer Benutzerschnittstelle sollte auf Basis eines Ordnungssystems erfolgen. Die Eigenschaften und Anordnung von Sichten sind wesentliche Kriterien in Bezug auf die Unterstützung der visuellen Kommunikation.
Problem	Im Unterschied zu konventionellen Benutzeroberflächen verändert sich die Gestalt des Interface, aufgrund des interaktiven Charakters der Bildräume, unentwegt. Die Anordnung von Sichten auf einer Darstellungsfläche muss weitgehend ohne Kenntnis über eine konkrete Ausgestaltung des Bildraums erfolgen.
Ebene	Konkretes Interfacedesign
Lösungsvorschlag	Der grundlegende Aufbau einer Benutzerschnittstelle wird, unter Zuhilfenahme von planaren Grundlinienrastern, festgelegt. Ein Ebenenraster beschreibt ein planares Grundlinienraster durch die Festlegung von Zeilen und Spalten. Auf diesen Ebenenrastern werden die Panels für die Sichten platziert. Die einzelnen Ebenenrastern bilden durch Platzierung und Schichtung Interfaceraster.
Vor- und Nachteile	<ul style="list-style-type: none"> <li>+ Flexible Erstellung von Ebenenrastern, wodurch umschließende Sichtenanordnungen oder in Teilen überlappende Formationen verwirklicht werden können</li> <li>+ Unterstützung der Aufmerksamkeitssteuerung durch die Entkopplung von Raster und Panel für eine Bereitstellung von dynamisch-veränderbaren multiplen Sichten</li> <li>+ Austausch, Wiederverwendbarkeit und Erweiterbarkeit von Rastern sind gegeben.</li> <li>+ Mithilfe des Entwurfsmusters können existierende Regeln und Hinweise aus den Bereichen der Print-, Comic- und der Webseitengestaltung integriert werden</li> <li>- Die Veränderung der Rastergestalt kann zu einer Neujustierung der Panels, respektive der Sichten führen und eine erhebliche Umstrukturierung der Sichten zur Folge haben.</li> <li>- Durch eine vorgegebene Einordnung von Panels in ein Raster ist eine vollständig individuelle Anordnung dieser nicht gegeben.</li> </ul>
Verwendung	Für Interfaces mit multiplen Sichten und bei situativen Sichteinblendungen (Pop-up-Sicht)
Verbindungen zu anderen Mustern	Durch die Einbettung von Bildräumen in ein Raster besteht eine enge Verbindung zur Sichtweise sowie zur Modifikation durch die Veränderungen eines Rasters zur Laufzeit

Tabelle 20: Kurzdarstellung des Entwurfsmusters Panelraster



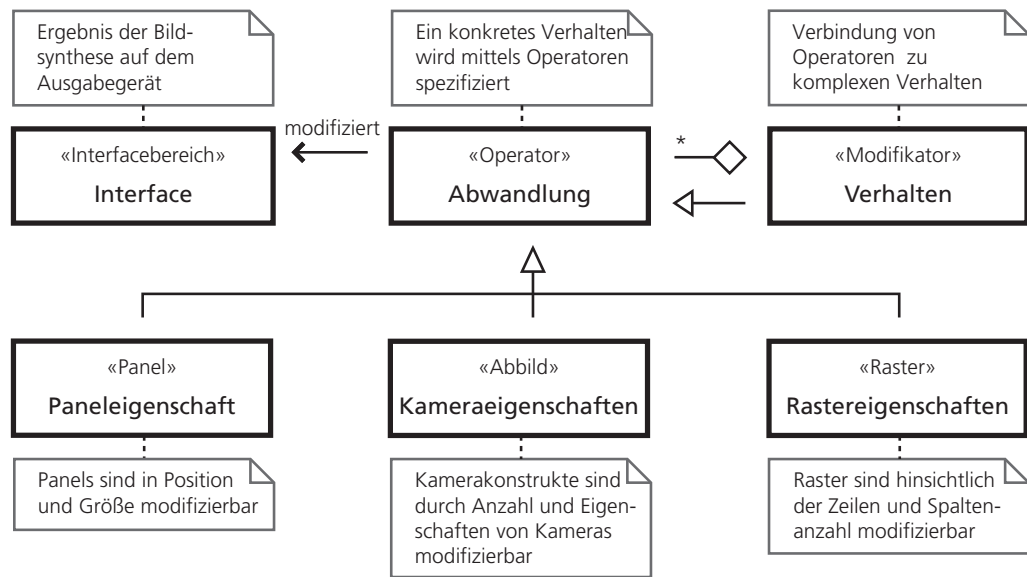
## Entwurfsmuster: Wandler



Merkmal	Erläuterung
Name	Wandler
Problemkontext	Interaktive 3D-Anwendungen bieten die Möglichkeit, die Darstellung des Objektraums während der Laufzeit zu verändern. Die dynamische Anpassbarkeit ist notwendig, damit durch die Veränderung des Erscheinungsbildes ein intendiertes Visualisierungsziel erreicht werden kann.
Problem	Die Vielgestaltigkeit des computergrafischen Bildraums resultiert im Entwurfsprozess in der Verfügbarkeit einer großen Anzahl an Darstellungsformen, die im interaktiven Prozess eine situative Anwendung finden. Diese Vielfalt erschwert die Konzeption eines Interface, da deren Verwirklichung, unter Berücksichtigung einer ständigen Veränderbarkeit der Darstellungsform zur Laufzeit der Anwendung, erfolgen muss.
Ebene	Konkretes Interfacedesign
Lösungsvorschlag	Die Flexibilität und situationsabhängige Anpassbarkeit der Darstellungsformen zur Gestaltung des Bildraums wird durch eine Trennung der visuellen Elemente des Objektraums von deren Abbildungsmethode erreicht. Eine flexible und bedingungs-basierte Zuordnung ermöglicht ferner die Visualisierung der Elemente im Bildraum.
Vor- und Nachteile	<ul style="list-style-type: none"> <li>+ Es wird eine individuelle Festlegung des Erscheinungsbildes einzelner Elemente durch dedizierte, wiederverwendbare Darstellungswandler erreicht.</li> <li>+ Die Erweiterbarkeit durch das Hinzufügen neuer Projektionsverfahren ist durch die Trennung von Darstellungsweisen und visueller Struktur gewährleistet.</li> <li>+ Die Elemente eines Objektraums können gemeinsam verwaltet werden, obgleich diese in einer Anwendung auf Basis unterschiedlicher Darstellungsweisen abgebildet werden</li> <li>+ Die konkrete visuelle Repräsentation eines Elementes kann zur Laufzeit dynamisch zugewiesen werden.</li> <li>- Bei der Anwendung des Entwurfsmusters ist eine erhöhte Ressourcenbelastung möglich, da Darstellungswandler zur Laufzeit reaktiviert, adaptiert und ausgeführt werden müssen.</li> </ul>
Verwendung	Für verzerrungsfreie oder positionsabhängige Detail-und-Kontext-Darstellungen
Verbindungen zu anderen Mustern	Das Muster weist eine Verbindung zum Sichtweise-Entwurfsmuster auf, da Darstellungsweisen dynamisch angepasst werden.

Tabelle 21: Kurzdarstellung des Entwurfsmusters Wandler

## Entwurfsmuster: Modifikation



Merkmal	Erläuterung
Name	Modifikation
Problemkontext	Die Integration neuer Verhaltenstechniken erfolgt häufig durch die Abwandlung oder die Spezialisierung existierender Techniken und in seltenen Fällen durch die vollständige Neuentwicklung eines Verhaltensbausteins und deren Aufnahme in die Menge existierender Formen. In der Konsequenz werden überwiegend etablierte Dialogtechniken in interaktiven Anwendungen eingesetzt.
Problem	Durch die Anwendung bestehender Verhaltenstechniken in originärer oder lediglich modifizierter Form wird der Designer in der Gestaltung von Benutzerschnittstellen eingeschränkt. Eine Neuentwicklung von Dialogtechniken hat wiederum für den Entwickler einen erheblichen Aufwand in der Umsetzung zur Folge.
Ebene	Konkretes Interfacedesign
Lösungsvorschlag	Ein flexibel veränderbares Verhalten wird auf Basis eines feineren Granularitätsgrades erreicht. Dazu wird eine Dialogtechnik in kombinierbare Operatoren zerlegt, deren Zusammensetzung in einen Modifikator im Entwurf konzipiert und während der Laufzeit einer Anwendung ausgeführt werden kann.
Vor- und Nachteile	<ul style="list-style-type: none"> <li>+ Eine strukturierte Erzeugung eines komplexen Verhaltens durch eine Abbildung von Einzeloperationen auf baumartige Strukturen.</li> <li>+ Eine leichte Erweiterbarkeit und Wiederverwendbarkeit ist durch die Integration neuer Operatoren gegeben</li> <li>+ Das Entwurfsmuster kann zur Definition des Verhaltens in einem Bildraum und zur Koordination verschiedener Sichten eingesetzt werden</li> <li>- Inkonsistenzen im Verhalten einer Schnittstelle werden durch das Entwurfsmuster nicht explizit aufgezeigt</li> <li>- Die Festlegung von Verhaltenseigenschaften als Abwandlung oder als Verhalten obliegt dem Nutzer des Musters in Abhängigkeit vom Verwendungszweck.</li> </ul>
Verwendung	Interaktive Anwendungen, die ein spezielles, flexibles oder unkonventionelles Verhalten aufweisen sollen
Verbindungen zu anderen Mustern	Durch das Entwurfsmuster werden Interfaces veränderbar, wodurch die Modifikation eine Verbindung zu den Mustern Sichtweise, Panelraster und Wandler besitzt.

Tabelle 22: Kurzdarstellung des Entwurfsmusters Modifikation

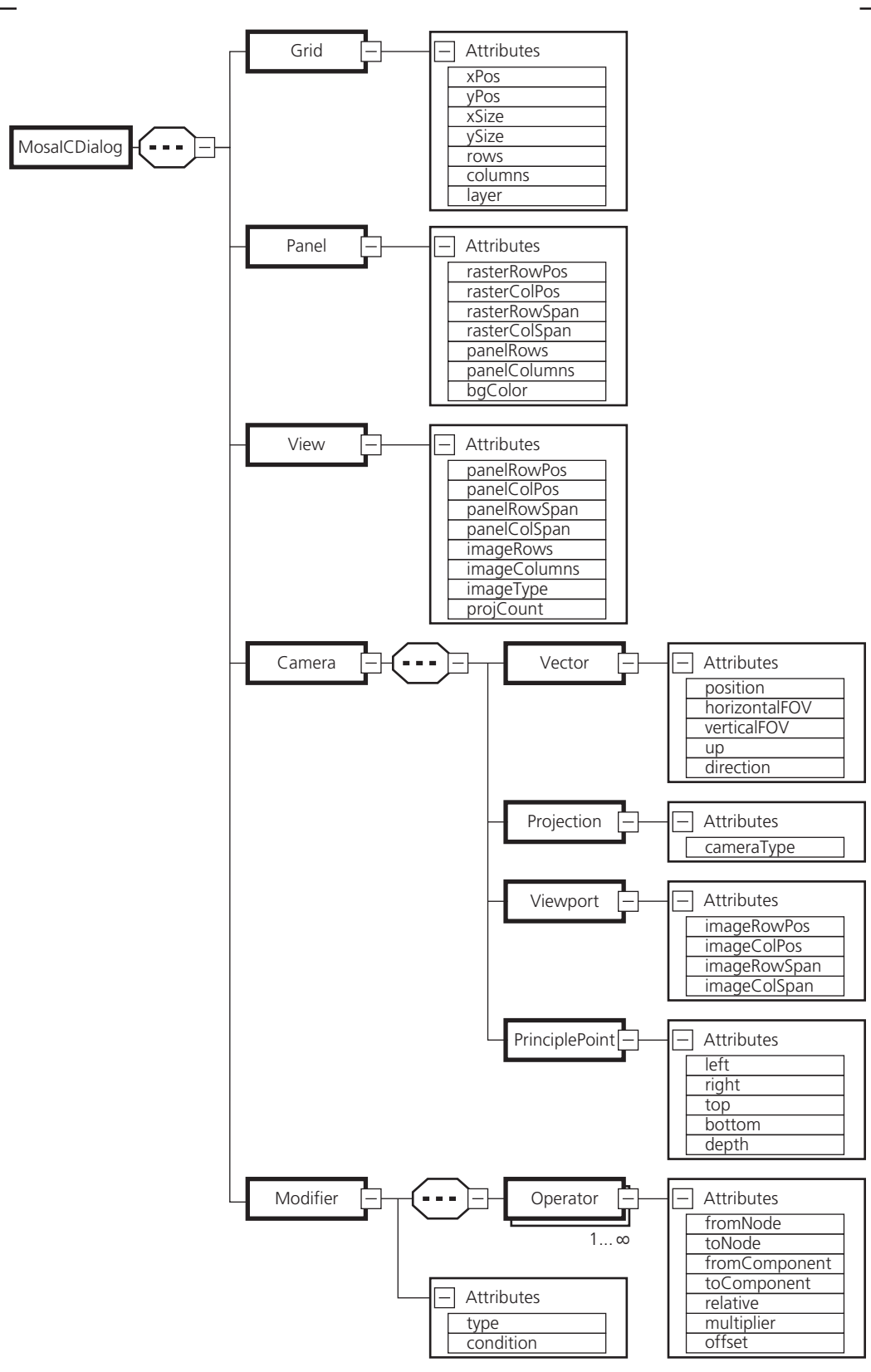


Abbildung 56: Struktur des MosalK:Dialog XML-Schema MosalK-Knoten

# Abkürzungsverzeichnis

API	Application Programming Interface
BiLL	Bildsprache LiveLab
CAD	Computer Aided Design
CAVE	Cave Automatic Virtual Environment
COLLADA	COLLABorative Design Activity
CORBA	Common Object Request Broker Architecture
DTD	Document Type Definition
FLTK	Fast Light ToolKit
GUI	Graphical User Interface
HMD	Head-Mounted Display
HUD	Head-up-Display
ISO	International Organization of Standardization
MBUID	Model Based User Interface Development
NPR	Non-Photorealistic Rendering
OSG	OpenSceneGraph
OSGi	Open Services Gateway initiative
OSP	Open Service Platform
PARC	Palo Alto Research Center (häufig Xerox PARC)
POCO	Portable Components
SDL	Scene Description Language
UIDL	User Interface Description Language
UIMS	User Interface Management Systems
UML	Unified Modeling Language
VRML	Virtual Reality Modeling Language
W3C	World Wide Web Consortium
WIMP	
X3D	Extensible 3D
XAML	Extensible Application Markup Language
XML	Extensible Markup Language
XSL(T)	Extensible Stylesheet Language (Transformations)
ZUI	Zoomable User Interface

# Webreferenzen

[@ACM]	ACM. Association for Computing Machinery's Special Interest Group on Computer Graphics, Online im Internet: <a href="http://www.siggraph.org/">http://www.siggraph.org/</a> , Stand: 2013-08-05.
[@AG-TV]	AG-TV, <i>AG Technische Visualistik - Projekt Bildsprache LiveLab</i> , Online im Internet: <a href="http://mq.inf.tu-dresden.de/redmine/projects/bill">http://mq.inf.tu-dresden.de/redmine/projects/bill</a> , Stand: 2012-08-30.
[@Alice]	ALICE. Alice - Lehr- Lernsoftware für Studenten, Online im Internet: <a href="http://www.alice.org/">http://www.alice.org/</a> , Stand: 2013-09-10.
[@Apple]	APPLE, <i>Apple Maps – Kartendienst des Unternehmens Apple</i> , Online im Internet: <a href="http://www.apple.com/de/ios/maps/">http://www.apple.com/de/ios/maps/</a> , Stand: 2013-06-10.
[@Applied Informatics]	APPLIED INFORMATICS, <i>Webseite der POCO C++ Bibliothek</i> , Online im Internet: <a href="http://pocoproject.org/">http://pocoproject.org/</a> , Stand: 2013-09-07.
[@Audacity-Team]	AUDACITY-TEAM, <i>Webseite der Open-Source-Software Audacity</i> , Online im Internet: <a href="http://audacity.sourceforge.net/?lang=de">http://audacity.sourceforge.net/?lang=de</a> , Stand: 2013-05-07.
[@Audi AG]	AUDI AG, <i>Audi Modell-Konfigurator</i> , Online im Internet: <a href="http://konfigurator.audi.de">http://konfigurator.audi.de</a> , Stand: 2013-09-03.
[@Autodesk]	AUTODESK, <i>Produktwebseite für die Software des Unternehmens Autodesk</i> , Online im Internet: <a href="http://www.autodesk.de/products">http://www.autodesk.de/products</a> , Stand: 2013-08-29.
[@BAW]	BAW, <i>Internetauftritt der Bundesanstalt für Wasserbau</i> , Online im Internet: <a href="http://www.baw.de/">http://www.baw.de/</a> , Stand: 2013-09-03.
[@CAIDA]	CAIDA, <i>Walrus - Visualisierungswerkzeug zur Darstellung von Graphen</i> , Online im Internet: <a href="http://www.caida.org/tools/visualization/walrus/">http://www.caida.org/tools/visualization/walrus/</a> , Stand: 2013-08-02.
[@Cameleon SIG]	CAMELEON SIG, <i>Projektwebseite des CAMELEON Projektes</i> , Online im Internet: <a href="http://girove.isti.cnr.it/projects/cameleon.html">http://girove.isti.cnr.it/projects/cameleon.html</a> , Stand: 2013-08-13.
[@Dassault Systèmes]	DASSAULT SYSTÈMES, <i>Unternehmenswebseite von Dassault Systèmes</i> , Online im Internet: <a href="http://www.3ds.com/">http://www.3ds.com/</a> , Stand: 2013-08-02.
[@Dice]	DICE, <i>Offizielle Webseite zum Spiel Battlefield der Firma DICE</i> , Online im Internet: <a href="http://www.battlefield.com/">http://www.battlefield.com/</a> , Stand: 2013-08-01.
[@FLTK]	FLTK, <i>Entwicklerwebseite des Fast Light ToolKit</i> , Online im Internet: <a href="http://www.fltk.org">http://www.fltk.org</a> , Stand: 2013-08-29.
[@Garmin]	GARMIN, <i>Internetauftritt der Firma Garmin Würzburg GmbH</i> , Online im Internet: <a href="http://www.navigon.com">http://www.navigon.com</a> , Stand: 2013-09-05.
[@Global Southend]	GLOBAL SOUTHEND, <i>Webseite des Spieleentwicklers</i> , Online im Internet: <a href="http://www.southend.se/games/game/xiii">http://www.southend.se/games/game/xiii</a> , Stand: 2013-08-22.
[@Google]	GOOGLE, <i>Webseite des Kartendienstes Google Earth</i> , Online im Internet: <a href="http://www.google.de/earth/index.html">http://www.google.de/earth/index.html</a> , Stand: 2013-09-01.
[@Irrlicht]	IRR LICHT. Irrlicht Engine - A free open source 3D engine, Online im Internet: <a href="http://irrlicht.sourceforge.net/">http://irrlicht.sourceforge.net/</a> , Stand: 2013-09-01.
[@Khronos Group]	KHRONOS GROUP, <i>Webseite der Collada Schema Definition</i> , Online im Internet: <a href="http://www.khronos.org">http://www.khronos.org</a> , Stand: 2013-09-04.

[@Linden Lab]	LINDEN LAB, <i>Second Life - Dresdner Gemäldegalerie Alte Meister</i> , Online im Internet: <a href="http://www.secondlife-neu-entdecken.de/projekte/kultur/dresden/">http://www.secondlife-neu-entdecken.de/projekte/kultur/dresden/</a> , Stand: 2013-09-01.
[@MeVisLab]	MEVISLAB, <i>Webseite des Unternehmens MeVisLab</i> , Online im Internet: <a href="http://www.mevislab.de/">http://www.mevislab.de/</a> , Stand: 2013-08-17.
[@Microsoft]	MICROSOFT, <i>MSDN: Microsoft Entwicklernetzwerk</i> , Online im Internet: <a href="http://msdn.microsoft.com/de-de/">http://msdn.microsoft.com/de-de/</a> , Stand: 2012-08-14.
[@NeoAxis]	NEOAXIS, <i>Webseite des Unternehmens NeoAxis</i> , Online im Internet: <a href="http://www.neoaxis.com/">http://www.neoaxis.com/</a> , Stand: 2013-09-02.
[@OGRE]	OGRE, <i>Webseite der Open Source 3D Graphics Engine OGRE</i> , Online im Internet: <a href="http://www.ogre3d.org/">http://www.ogre3d.org/</a> , Stand: 2013-09-04.
[@OMG]	OMG, <i>Webseite der Object Management Group for Modeling and Metadata Spezifikation</i> , Online im Internet: <a href="http://www.omg.org/spec/index.htm">http://www.omg.org/spec/index.htm</a> , Stand: 2013-08-29.
[@PLML]	PLML, <i>Webseite der erweiterten Entwurfsmuster-Auszeichnungssprache PLMLx</i> , Online im Internet: <a href="http://www.cs.kent.ac.uk/people/staff/saf/patterns/diethelm/plmlx_doc/plml_doc.dtd.html">http://www.cs.kent.ac.uk/people/staff/saf/patterns/diethelm/plmlx_doc/plml_doc.dtd.html</a> , Stand: 2013-08-20.
[@POV-Ray]	POV-RAY, <i>Internetseite des POV-Ray Renderers</i> , Online im Internet: <a href="http://www.povray.org/">http://www.povray.org/</a> , Stand: 2012-08-29
[@Rockstar Games]	ROCKSTAR GAMES, <i>Die offizielle Webseite von Grand Theft Auto</i> , Online im Internet: <a href="http://www.rockstargames.de/grandtheftauto/">http://www.rockstargames.de/grandtheftauto/</a> , Stand: 2013-04-01.
[@Skyrails]	SKYRAILS, <i>Blog zum Projekt Skyrails</i> , Online im Internet: <a href="http://www.visualcomplexity.com/vc/project.cfm?id=511">http://www.visualcomplexity.com/vc/project.cfm?id=511</a> , Stand: 2011-09-09.
[@SOF]	SOF, <i>Entwicklerwebseite für das Service Oriented Framework</i> , Online im Internet: <a href="http://sof.tiddlyspot.com/">http://sof.tiddlyspot.com/</a> , Stand: 2012-09-04.
[@VLC]	VLC, <i>Offizielle Webseite des VLC Players</i> , Online im Internet: <a href="http://www.videolan.org/">http://www.videolan.org/</a> , Stand: 2013-09-06.
[@W3C]	W3C, <i>Internetauftritt des World Wide Web Konsortiums</i> , Online im Internet: <a href="http://www.w3.org/">http://www.w3.org/</a> , Stand: 2013-09-03.
[@Web3D]	WEB3D, <i>Internetauftritt des Web3D Konsortiums</i> , Online im Internet: <a href="http://www.web3d.org">http://www.web3d.org</a> , Stand: 2013-08-14.
[@XML]	XML, <i>Webseite der Arbeitsgruppe der Extensible Markup Language (XML)</i> , Online im Internet: <a href="http://www.w3.org/XML/">http://www.w3.org/XML/</a> , Stand: 2013-09-03.
[@XSL]	XSL, <i>Webseite der Arbeitsgruppe der Extensible Stylesheet Language Family (XSL)</i> , Online im Internet: <a href="http://www.w3.org/Style/XSL/">http://www.w3.org/Style/XSL/</a> , Stand: 2013-09-03.

# Literaturverzeichnis

- [AGARWALA U. A. 2006] AGARWALA, A., AGRAWALA, M., COHEN, M., SALESIN, D. UND SZELISKI, R. (2006): Photographing Long Scenes with Multi-Viewpoint Panoramas, In: *ACM Transactions on Graphics*, Vol.25/3, S. 853–861.
- [AGRAWALA & STOLTE 2001] AGRAWALA, M. UND STOLTE, C. (2001): Rendering Effective Route Maps: Improving Usability Through Generalization, In: *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH'01)*, ACM Press, S. 241–249.
- [AGRAWALA U. A. 2000] AGRAWALA, M., ZORIN, D. UND MUNZNER, T. (2000): Artistic Multiprojection Rendering, In: *Proceedings of the Eurographics Workshop on Rendering Techniques 2000 (EGWR'00)*, Springer-Verlag, S. 125–136.
- [AKENINE-MÖLLER U. A. 2008] AKENINE-MÖLLER, T., HAINES, E. UND HOFFMAN, N. (2008): *Real-Time Rendering*, A.K. Peters Ltd.
- [ALBERTI 1972] ALBERTI, L.B. (1972): *On Painting and On Sculpture: The Latin Texts of De Pictura and De Statua, edited with Translations, Introduction and Notes*, Phaidon Press.
- [ALEXANDER U. A. 1977] ALEXANDER, C., ISHIKAWA, S. UND SILVERSTEIN, M. (1977): *A Pattern Language: Towns, Buildings, Construction*, Oxford University Press.
- [ALIAS SYSTEMS CORPORATION 2005] ALIAS SYSTEMS CORPORATION. (2005): *SDL: Scene Description Language*, Dokumentation zu Studio Tools 13, Toronto, Canada.
- [AMAR & STASKO 2004] AMAR, R. UND STASKO, J. (2004): A Knowledge Task-Based Framework for Design and Evaluation of Information Visualizations, In: *IEEE Symposium on Information Visualization, 2004 (INFOVIS'04)*, IEEE Computer Society Press, S. 143–150.
- [ANDRIENKO & ANDRIENKO 2007] ANDRIENKO, G. UND ANDRIENKO, N. (2007): Coordinated Multiple Views: a Critical View, In: *Proceedings of the Fifth International Conference on Coordinated and Multiple Views in Exploratory Visualization (CMV'07)*, IEEE Computer Society Press, S. 72–74.
- [ANDRIENKO & ANDRIENKO 2006] ANDRIENKO, N. UND ANDRIENKO, G. (2006): *Exploratory Analysis of Spatial and Temporal Data: A Systematic Approach*, Springer Verlag.
- [ANGEL 2005] ANGEL, E. (2005): *Interactive Computer Graphics: A Top-Down Approach with OpenGL*, Addison-Wesley.
- [ARK U. A. 1998] ARK, W., CHRISTOPHER, D., TED, D. UND ZHAI, S.S. (1998): Representation Matters: The Effect of 3D Objects and a Spatial Metaphor in a Graphical User Interface, In: *Proceedings of HCI on People and Computers XIII (HCI'98)*, Springer-Verlag, S. 209–219.
- [ARNHEIM 2003] ARNHEIM, R. (2003): *Die Macht der Mitte: Eine Kompositionslehre für die bildenden Künste*, Dumont Buchverlag.
- [ABMANN 2003] ABMANN, U. (2003): *Invasive Software Composition*, Springer Verlag.
- [BALLENDAT U. A. 2010] BALLENDAT, T., MARQUARDT, N. UND GREENBERG, S. (2010): Proxemic Interaction: Designing for a Proximity and Orientation-Aware Environment, In: *ACM International Conference on Interactive Tabletops and Surfaces (ITS'10)*, ACM Press, S. 121–130.
- [BARBOSA & PAULA 2003] BARBOSA, S.D.J. UND PAULA, M.G. (2003): Interaction Modelling as a Binding Thread in the Software Development Process, In: *Proceedings of the Workshop on Bridging the Gaps Between Software Engineering and Human-Computer Interaction (ICSE'03)*, Springer-Verlag, S. 84–91.
- [BAYARRI 1995] BAYARRI, S. (1995): Computing Non-Planar Perspectives in Real Time, In: *Computers & Graphics*, Vol.19/3, S. 431–440.



- [BEALE & BORDBAR 2005] BEALE, R. UND BORDBAR, B. (2005): Using modelling to put HCI design patterns to work, In: *HCI International. 11th International Conference on Human-Computer Interaction, Las Vegas, Nevada, USA*.
- [BEAUDOUIN-LAFON 2004] BEAUDOUIN-LAFON, M. (2004): Designing Interaction, not Interfaces, In: *Proceedings of the working conference on Advanced Visual Interfaces (AVI'04)*, ACM Press, S. 15–22.
- [BEHRENS U. A. 2010] BEHRENS, A., GUSKI, M., STEHLE, T., GROSS, S. UND AACH, T. (2010): A non-linear multi-scale blending algorithm for fluorescence bladder images, In: *Computer Science - Research and Development*, Vol.26/1-2, S. 125–134.
- [BERTIN 1981] BERTIN, J. (1981): *Graphics and Graphic Information Processing*, De Gruyter.
- [BERTOLINE 1998] BERTOLINE, G.R. (1998): Visual Science: An Emerging Discipline, In: *Journal for Geometry and Graphics*, Vol.2/2, S. 181–187.
- [BIERBAUM U. A. 2001] BIERBAUM, A., JUST, C., HARTLING, P., MEINERT, K., BAKER, A. UND CRUZ-NEIRA, C. (2001): VR Juggler: A Virtual Platform for Virtual Reality Application Development, In: *Proceedings of the Virtual Reality 2001 Conference (VR'01)*, IEEE Computer Society Press, S. 89–96.
- [BIERMANN & JUNG 2004] BIERMANN, P. UND JUNG, B. (2004): Variant design in immersive virtual reality: A markup language for scalable CSG parts, In: *Articulated Motion and Deformable Objects*, S. 123–133.
- [BLANCHETTE & SUMMERFIELD 2008] BLANCHETTE, J. UND SUMMERFIELD, M. (2008): *C++ GUI Programming with Qt 4*, Prentice Hall.
- [BORCHERS 2001] BORCHERS, J. (2001): *A Pattern Approach to Interaction Design*, Wiley.
- [BORCHERS 2000A] BORCHERS, J.O. (2000): Interaction Design Patterns: Twelve Theses, In: *Workshop on Pattern Languages for Interaction Design (CHI'00)*, ACM Press, S. 1–6.
- [BORCHERS 2000B] BORCHERS, J.O. (2000): A Pattern Approach to Interaction Design, In: *Proceedings of the 3rd Conference on Designing Interactive Systems (DIS'00)*, ACM Press, S. 114–131.
- [BORECZKY U. A. 2000] BORECZKY, J., GIRGENSOHN, A., GOLOVCHINSKY, G. UND UCHIHASHI, S. (2000): An Interactive Comic Book Presentation for Exploring Video, In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI'00)*, ACM Press, S. 185–192.
- [BÖTTGER U. A. 2008] BÖTTGER, J., PREISER, M., BALZER, M. UND DEUSSEN, O. (2008): Detail-In-Context Visualization for Satellite Imagery, In: *Computer Graphics Forum*, Vol.27/2, S. 587–596.
- [BOWMAN U. A. 2006] BOWMAN, D.A., CHEN, J., WINGRAVE, C.A., LUCAS, J., RAY, A., POLYS, N.F., LI, Q., HACIAHMETOGLU, Y., KIM, J.S., KIM, S., BOEHRINGER, R. UND NI, T. (2006): New Directions in 3D User Interfaces, In: *The International Journal of Virtual Reality*, Vol.28/2, S. 3–14.
- [BOWMAN U. A. 2008] BOWMAN, D.A., COQUILLART, S., FROELICH, B., HIROSE, M., KITAMURA, Y., KIYOKAWA, K. UND STUERZLINGER, W. (2008): 3D User Interfaces: New Directions and New Perspectives, In: *IEEE Computer Graphics and Applications*, Vol.28/6, S. 20–36.
- [BOWMAN & HODGES 1997] BOWMAN, D.A. UND HODGES, L.F. (1997): An Evaluation of Techniques for Grabbing and Manipulating Remote Objects in Immersive Virtual Environments, In: *Proceedings of the 1997 Symposium on Interactive 3D Graphics (I3D'97)*, ACM Press, S. 35–40.
- [BOWMAN U. A. 2004] BOWMAN, D.A., KRUIFF, E., LAVIOLA, J.J. UND POUPYREV, I. (2004): 3D User Interfaces: Theory and Practice, In: *Recherche*, Vol.67, S. 02.
- [BOWMAN U. A. 2012] BOWMAN, D.A., MCMAHAN, R.P. UND RAGAN, E.D. (2012): Questioning Naturalism in 3D User Interfaces, In: *Communications of the ACM*, Vol.55/9, S. 78–88.

- [BRADEN & HORTIN 1982] BRADEN, R.A. UND HORTIN, J.A. (1982): Identifying the Theoretical Foundations of Visual Literacy, In: *Television and Visual Literacy. Readings from the 13th Annual Conference of the International Visual Literacy Association*. Bloomington Indiana University.
- [BROSZ U. A. 2007] Brosz, J., SAMAVATI, F.F., SHEELAGH, M.T.C. UND SOUSA, M.C. (2007): Single Camera Flexible Projection, In: *Proceedings of the 5th international Symposium on Non-Photorealistic Animation and Rendering* (NPAR'07), ACM Press, S. 33–42.
- [BRUTZMAN & DALY 2007] BRUTZMAN, D. UND DALY, L. (2007): *X3D: Extensible 3D Graphics for Web Authors*, Morgan Kaufmann.
- [BUDDE U. A. 1988] BUDDE, U., DRECHSLER, W. UND HUBER-RUPPEL, M. (1988): *Wege zur Abstraktion. 80 Meisterwerke aus der Sammlung Thyssen-Bornemisza*, Edition Cantz.
- [BÜTTNER 2005] BÜTTNER, F. (2005): Der Blick auf das Bild. Betrachter und Perspektive in der Renaissance, In: *Anblick und Augenblick: Ein interdisziplinäres Symposium*. Königshausen & Neumann, S. 131–163.
- [CALVARY U. A. 2003] CALVARY, G., COUTAZ, J., THEVENIN, D., LIMBOURG, Q., BOUILLON, L. UND VANDERDONCKT, J. (2003): A unifying reference framework for multi-target user interfaces, In: *Interacting with Computers*, Vol.15/3, S. 289 – 308.
- [CARD U. A. 1999] CARD, S.K., MACKINLAY, J.D. UND SHNEIDERMAN, B. (1999): *Readings in Information Visualization: Using Vision to Think*, Morgan Kaufmann.
- [CARPENDALE U. A. 1996] CARPENDALE, M.S.T., COWPERTHWAIT, D.J. UND FRACCHIA, F.D. (1996): Distortion Viewing Techniques for 3-Dimensional Data, In: *Proceedings of the 1996 IEEE Symposium on Information Visualization* (INFOVIS'96), IEEE Computer Society Press, S. 46–53.
- [CARROLL U. A. 2010] CARROLL, R., AGARWALA, A. UND AGRAWALA, M. (2010): Image Warps for Artistic Perspective Manipulation, In: *ACM Transactions on Graphics*, Vol.29/4, S. 127:1–127:9.
- [CARROLL U. A. 2009] CARROLL, R., AGRAWALA, M. UND AGARWALA, A. (2009): Optimizing Content-Preserving Projections for Wide-Angle Images, In: *ACM Transactions on Graphics*, Vol.28/3, S. Article 43.
- [CEDILNIK U. A. 2006] CEDILNIK, A., GEVECI, B., MORELAND, K., AHRENS, J. UND FAVRE, J. (2006): Remote Large Data Visualization in the ParaView Framework, In: *Proceedings of the Eurographics Symposium on Parallel Graphics and Visualization* (EGPGV'06), S. 162–170.
- [CHEN 2005] CHEN, C. (2005): Top 10 Unsolved Information Visualization Problems, In: *IEEE Computer Graphics and Applications*, Vol.25/4, S. 12–16.
- [CHIN 2002] CHIN, R. (2002): Three-Dimensional File System Browser, In: *Crossroads*, Vol.9/1, S. 16–18.
- [CHLEBEK 2002] CHLEBEK, P. (2002): *Spezifikation von Benutzeroberflächen: Ausgangssituation, Bedarf, Umfang, Sprachmittel und Praxiseinführung der formalen Oberflächenspezifikation für dialogintensive Anwendungen*, Praxisbericht von Eos Projectpeople, Eos Projectpeople, München, Deutschland.
- [COCKBURN & MCKENZIE 2002] COCKBURN, A. UND MCKENZIE, B. (2002): Evaluating the Effectiveness of Spatial Memory in 2D and 3D Physical and Virtual Environments, In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI'02), ACM Press, S. 203–210.
- [COCONU U. A. 2006] COCONU, L., DEUSSEN, O. UND HEGE, H.C. (2006): Real-Time Pen-and-Ink Illustration of Landscapes, In: *Proceedings of the 4th International Symposium on Non-Photorealistic Animation and Rendering* (NPAR'06), ACM Press, S. 27–35.

- [COLEMAN U. A. 2005] COLEMAN, P., SINGH, K., BARRETT, L., SUDARSANAM, N. UND GRIMM, C. (2005): 3D Screen-space Widgets for Non-linear Projection, In: *Proceedings of the 3rd International Conference on Computer Graphics and Interactive Techniques in Australasia and South East Asia* (AUIC'05), ACM Press, S. 221–228.
- [CONSTANTINE & LOCKWOOD 1999] CONSTANTINE, L.L. UND LOCKWOOD, L.A.. (1999): *Software for Use: A Practical Guide to the Models and Methods of Usage-Centered Design*, Addison-Wesley.
- [CONVERTINO U. A. 2003] CONVERTINO, G., CHEN, J., YOST, B., RYU, Y.-S. UND NORTH, C. (2003): Exploring Context Switching and Cognition in Dual-View Coordinated Visualizations, In: *Fifth International Conference on Coordinated and Multiple Views in Exploratory Visualization* (CMV'03), IEEE Computer Society Press, S. 55–62.
- [CRAFT & CAIRNS 2005] CRAFT, B. UND CAIRNS, P. (2005): Beyond Guidelines: What Can We Learn from the Visual Information Seeking Mantra?, In: *Proceedings of the Ninth International Conference on Information Visualization, 2005* (IV'05), IEEE Computer Society Press, S. 110 – 118.
- [CRISTESCU 2009] CRISTESCU, V. (2009): The Reverse Perspective in the Orthodox Iconography according to P. Florenski, A Dogmatic Perspective, In: *European Journal of Science and Theology*, Vol.5/1, S. 41–50.
- [CUTTING 1997] CUTTING, J.E. (1997): How the eye measures reality and virtual reality, In: *Behavior Research Methods Instruments and Computers*, Vol.29/1, S. 27–36.
- [DACHSELT 2001] DACHSELT, R. (2001): CONTIGRA - Towards a Document-based Approach to 3D Components, In: *Design of Virtual Environments and 3D-Components' at the ACM Web3D 2001 Symposium* (Web3D'01).
- [DACHSELT 2004] DACHSELT, R. (2004): *Eine deklarative Komponentenarchitektur und Interaktionsbausteine für dreidimensionale multimediale Anwendungen*, Der Andere Verlag.
- [DACHSELT U. A. 2007] DACHSELT, R., FIGUEROA, P., LIND, I. UND BROLL, W. (2007): Mixed Reality User Interfaces: Specification, Authoring, Adaptation, In: *Workshop Proceedings: 2nd International Workshop at the IEEE Virtual Reality 2007 Conference* (MRUI'07), IEEE Computer Society Press.
- [DARKEN & SIBERT 1993] DARKEN, R.P. UND SIBERT, J.L. (1993): A Toolset for Navigation in Virtual Environments, In: *Proceedings of the 6th Annual ACM Symposium on User Interface Software and Technology* (UIST'93), ACM Press, S. 157–165.
- [DÄBLER 1999] DÄBLER, R. (1999): Informationsvisualisierung: Stand, Kritik und Perspektiven, In: *Methoden/Strategien der Visualisierung in Medien, Wissenschaft und Kunst*. Wissenschaftlicher Verlag Trier.
- [DÄBLER & PALM 1998] DÄBLER, R. UND PALM, H. (1998): *Virtuelle Informationsräume mit VRML: Informationen recherchieren und präsentieren in 3D*, dpunkt-Verlag.
- [DEUSSEN 2001] DEUSSEN, O. (2001): Nichtrealistische Computergraphik, In: *Informatik-Spektrum*, Vol.24/2, S. 71–80.
- [DIN 2002] DIN. (2002): 14915: Software-Ergonomie für Multimedia-Benutzungsschnittstellen–Teil 1: Gestaltungsgrundsätze und Rahmenbedingungen (ISO 14915-1: 2002), In: *Berlin: Beuth*.
- [DIX U. A. 2003] DIX, A., FINLAY, J.E., ABOWD, G.D. UND BEALE, R. (2003): *Human-Computer Interaction*, Prentice Hall.
- [DURAND 2001] DURAND, F. (2001): Non-linear Drawing systems, *Vortragsreihe „The Art and Science of Depiction“*, Massachusetts Institute of Technology, Cambridge, USA.

- [DURAND 2002A] DURAND, F. (2002): An Invitation to Discuss Computer Depiction, In: *Proceedings of the 2nd international Symposium on Non-photorealistic Animation and Rendering (NPAR'02)*, ACM Press, S. 111–124.
- [DURAND 2002B] DURAND, F. (2002): Limitations of the medium and pictorial techniques, In: *Perceptual and Artistic Principles for Effective Computer Depiction, Course Notes for ACM SIGGRAPH 2002*, ACM Press, S. 27–45.
- [DÜRER 1525] DÜRER, A. (1525): *Underweysung der Messung, mit dem Zirckel und Richtscheyt, in Linien, Ebenen unnd gantzen Corporen*, Hieronymus Andreae.
- [DUYNE U. A. 2003] DUYNÉ, D.K.V., LANDAY, J.A. UND HONG, J.I. (2003): *The Design of Sites: Patterns, Principles, and Processes for Crafting a Customer-Centered Web Experience*, Addison-Wesley.
- [EBNER 2007] EBNER, T. (2007): *BiLL (Bildsprache LiveLab) Konzeption und Realisierung einer interaktiven Arbeitsumgebung für die Erforschung wahrnehmungsrealistischer Projektion*, Diplomarbeit, Technische Universität Dresden, Dresden, Deutschland.
- [VON EINEM 1967] VON EINEM, H. (1967): *Masaccios Zinsgroschen*, Westdeutscher Verlag.
- [EISNER 2001] EISNER, W. (2001): *Comics & Sequential Art: Principles & Practice of the World's Most Popular Art Form*, Poorhouse Press.
- [ELMQVIST 2005] ELMQVIST, N. (2005): BalloonProbe: Reducing Occlusion in 3D using Interactive Space Distortion, In: *Proceedings of the ACM symposium on Virtual Reality Software and Technology (VRST'05)*, ACM Press, S. 134–137.
- [ERBSMEHL & HANNAWALD 2008] ERBSMEHL, C. UND HANNAWALD, L. (2008): Simulation realer Unfälleinlaufszzenarien der German In-Depth Accident Study (GIDAS), In: *Tagungsband der VDI Tagung 2008*, S. 1–14.
- [ERNST & LOCHER 1983] ERNST, B. UND LOCHER, J. (1983): *The Magic Mirror of M.C. Escher*, Harry N. Abrams, Inc.
- [EYSENCK & KEANE 2010] EYSENCK, M.W. UND KEANE, M.T. (2010): *Cognitive psychology: A student's handbook*, Taylor & Francis Ltd.
- [FELDT U. A. 2005] FELDT, N., PETTERSSON, H., JOHANSSON, J. UND JERN, M. (2005): Tailor-made Exploratory Visualization for Statistics Sweden, In: *Third International Conference on Coordinated and Multiple Views in Exploratory Visualization (CMV'05)*, IEEE Computer Society Press, S. 133–142.
- [FIGUEROA U. A. 2002] FIGUEROA, P., GREEN, M. UND HOOVER, H.J. (2002): InTml: A Description Language for VR Applications, In: *Proceedings of the Seventh International Conference on 3D Web Technology (Web3D'02)*, ACM Press, S. 53–58.
- [FINCHER & FINLAY 2003] FINCHER, S. UND FINLAY, J. (2003): Perspectives on HCI Patterns: Concepts and Tools, In: *Interfaces*, Vol.56, S. 26–28.
- [FISCHER 1996] FISCHER, L. (1996): Perspektive und Rahmung. Zur Geschichte einer Konstruktion von „Natur“, In: *Die Mobilisierung des Sehens. Zur Vor- und Frühgeschichte des Films in Literatur und Kunst*. S. 69–96.
- [FLORENSKIJ 1997] FLORENSKIJ, P.A. (1997): *Raum und Zeit*, KONTEXTverlag.
- [FLORENSKIJ 2000] FLORENSKIJ, P.A. (2000): *Iconostasis*, Oakwood Publications.
- [FLUSSER 1999] FLUSSER, V. (1999): *Ins Universum der technischen Bilder*, European Photography.
- [FOLEY 1995] FOLEY, J.D. (1995): *Computer Graphics: Principles and Practice*, Addison-Wesley.

- [FOLMER U. A. 2003] FOLMER, E., VAN GURP, J. UND BOSCH, J. (2003): A Framework for Capturing the Relationship between Usability and Software Architecture, In: *Software Process: Improvement and Practice*, Vol.8/2, S. 67–87.
- [FORBRIG 2002] FORBRIG, P. (2002): *Objektorientierte Softwareentwicklung mit UML*, Fachbuchverlag Leipzig.
- [FOWLER 2004] FOWLER, M. (2004): *UML distilled*, Addison-Wesley.
- [FRANKE & GROH 2006] FRANKE, I.S. UND GROH, R. (2006): Colour Perspective in context of Navigation through Virtual Worlds, In: *Proceedings of The Virtual 2006 - Designing Digital Experience*, M3 Research Platform.
- [FRANKE U. A. 2008] FRANKE, I.S., PANNASCH, S., HELMERT, J.R., RIEGER, R., GROH, R. UND VELICHKOVSKY, B.M. (2008): Towards Attention-Centered Interfaces: An Aesthetic Evaluation of Perspective with Eye Tracking, In: *ACM Transactions on Multimedia Computing, Communications, and Applications*, Vol.4/3, S. 1–13.
- [FRANKE U. A. 2007] FRANKE, I.S., ZAVESKY, M. UND DACHSELT, R. (2007): Learning from Painting: Perspective-dependent Geometry Deformation for Perceptual Realism, In: *Proceedings of the 13th Eurographics Symposium on Virtual Environments (EGVE'07)*, IPT-EGVE, S. 117–120.
- [FRANKE U. A. 2005] FRANKE, I.S., ZAVESKY, M. UND SCHINDLER, J. (2005): Multiperspektive versus Ergonomie, In: *50. Internationales Wissenschaftliches Kolloquium (IWK'05)*, Technische Universität Illmenau, S. 488–501.
- [FUJISHIRO U. A. 2000] FUJISHIRO, I., ICHIKAWA, Y., FURUHATA, R. UND TAKESHIMA, Y. (2000): GADGET/IV: A Taxonomic Approach to Semi-Automatic Design of Information Visualization Applications Using Modular Visualization Environment, In: *Proceedings of the IEEE Symposium on Information Visualization (InfoVis'00)*, IEEE Computer Society Press, S. 77–83.
- [FUJISHIRO U. A. 1997] FUJISHIRO, I., TAKESHIMA, Y., ICHIKAWA, Y. UND NAKAMURA, K. (1997): GADGET: Goal-Oriented Application Design Guidance for Modular Visualization Environments, In: *Proceedings on Visualization'97 (VIS'97)*, IEEE Computer Society Press, S. 245–252.
- [GAMMA U. A. 1994] GAMMA, E., HELM, R., JOHNSON, R. UND VLISSIDES, J. (1994): *Design Patterns: Elements of Reusable Object-Oriented Software*, Addison-Wesley.
- [GEYER 1994] GEYER, B. (1994): *Scheinwelten. Die Gesichte der Perspektive.*, E. A. Seemann.
- [GLAESER & GRÖLLER 1999] GLAESER, G. UND GRÖLLER, E. (1999): Fast generation of curved perspectives for ultra-wide-angle lenses in VR applications, In: *The Visual Computer*, Vol.15/7-8, S. 365–376.
- [GLASSNER 1989] GLASSNER, A.S. (1989): *An Introduction to Ray Tracing*, Morgan Kaufmann.
- [GOLDSTEIN 2007] GOLDSTEIN, E.B. (2007): *Cognitive Psychology: Connecting Mind, Research, and Everyday Experience*, Wadsworth Publishing.
- [GOMBRICH 1960] GOMBRICH, E.H. (1960): *Art and Illusion*, Phaidon Press.
- [GOMBRICH 1996] GOMBRICH, E.H. (1996): *Die Geschichte der Kunst*, S. Fischer.
- [GOOCH U. A. 1998] GOOCH, A., GOOCH, B., SHIRLEY, P. UND COHEN, E. (1998): A Non-Photorealistic Lighting Model For Automatic Technical Illustration, In: *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH'98)*, ACM Press, S. 447–452.
- [GOOCH & GOOCH 2001] GOOCH, B. UND GOOCH, A. (2001): *Non-Photorealistic Rendering*, A. K. Peters Ltd.
- [GRAHAM 2003] GRAHAM, I. (2003): *A pattern language for Web usability*, Addison-Wesley.

- [GRAU 2003] GRAU, O. (2003): *Virtual Art : From Illusion to Immersion*, MIT Press.
- [GREEN & JACOB 1991] GREEN, M. UND JACOB, R. (1991): SIGGRAPH '90 Workshop report: software architectures and metaphors for non-WIMP user interfaces, In: *ACM SIGGRAPH Computer Graphics*, Vol.25/3, S. 229–235.
- [GREEN U. A. 1999] GREEN, S., SALESIN, D., SCHOFIELD, S., HERTZMANN, A., LITWINOWICZ, P., GOOCH, A., CURTIS, C. UND GOOCH, B. (1999): Non-photorealistic rendering, In: *SIGGRAPH 99 - Non-Photorealistic Rendering Course Notes* (SIGGRAPH '99), ACM Press.
- [GROH 2007] GROH, R. (2007): *Das Interaktions-Bild Theorie und Methodik der Interfacegestaltung*, TUDpress.
- [GROH 2008] GROH, R. (2008): Vom Operieren und Orientieren - zu den Grundformen der Interaktion in 3D-Szenen, In: *Industriedesign und Ingenieurwissenschaften technisches Design in Forschung, Lehre und Praxis* (Symposium Technisches Design), TUDpress, S. 95–106.
- [GROH 2011] GROH, R. (2011): *Wieder mehr Sehen! Aktuelle Einblicke in die Technische Visualistik*, TUDpress.
- [HEER & AGRAWALA 2006] HEER, J. UND AGRAWALA, M. (2006): Software Design Patterns for Information Visualization, In: *IEEE Transactions on Visualization and Computer Graphics*, Vol.12/5, S. 853–860.
- [HERCZEG 2006] HERCZEG, M. (2006): *Interaktionsdesign : Gestaltung interaktiver und multimedialer Systeme*, Oldenbourg Wissenschaftsverlag.
- [HOCKNEY 2006] HOCKNEY, D. (2006): *Secret Knowledge: Rediscovering the Lost Techniques of the Old Masters*, Thames & Hudson.
- [HOFMANN 1998] HOFMANN, W. (1998): *Die Moderne im Rückspiegel: Hauptwege der Kunstgeschichte*, C.H.Beck.
- [HOLT 2009] HOLT, J. (2009): *A Pragmatic Guide to Business Process Modelling*, British Computer Society.
- [HSU U. A. 2011] HSU, W.H., MA, K.L. UND CORREA, C. (2011): A Rendering Framework for Multiscale Views of 3D Models, In: *Proceedings of the 2011 SIGGRAPH Asia Conference (SA'11)*, ACM Press, S. 131:1–131:10.
- [JACOB U. A. 1999] JACOB, R.J.K., DELIGIANNIDIS, L. UND MORRISON, S. (1999): A Software Model and Specification Language for Non-WIMP User Interfaces, In: *ACM Transactions on Computer-Human Interaction*, Vol.6/1, S. 1–46.
- [JOHNSON 2004] JOHNSON, C. (2004): Top Scientific Visualization Research Problems, In: *IEEE Computer Graphics and Applications*, Vol.24/4, S. 13–17.
- [DE JONG & KLINT 2003] DE JONG, H. UND KLINT, P. (2003): ToolBus: The Next Generation, In: *Formal Methods for Components and Objects*. Springer-Verlag, S. 220–241.
- [JUTZ & SCHLEMMER 1989] JUTZ, G. UND SCHLEMMER, G. (1989): Zur Geschichtlichkeit des Blicks, In: *Sprung im Spiegel. Filmisches Wahrnehmen zwischen Fiktion und Wirklichkeit*. Sonderzahl Verlag, S. 15–32.
- [KAMMER 2009] KAMMER, D. (2009): *Neukonzeption einer erweiterbaren Software-Architektur für ein echtzeitfähiges Visualisierungssystem*, Diplomarbeit, Technische Universität Dresden, Dresden, Deutschland.
- [KAMMER U. A. 2012] KAMMER, D., WOJDCIAK, J., EBNER, T., FRANKE, I.S. UND GROH, R. (2012): A component-oriented framework for experimental computer graphics, In: *Elsevier - Computer Standards & Interfaces*, Vol.34/1, S. 93–100.
- [KAMMER U. A. 2013] KAMMER, D., WOJDCIAK, J. UND GROH, R. (2013): Interactive Color Perspective for 3D Graphics Applications: Enhancing Depth Perception and the Understanding of Object Relations, In: *Communications in Computer and Information Science Volume 373 (HCI'13)*, S. 483–487.



- [KAPOR 1996] KAPOR, M. (1996): A Software Design Manifesto, In: *Bringing Design to Software*. ACM Press, S. 3–11.
- [KATO U. A. 2000] KATO, H., BILLINGHURST, M., POUPYREV, I., IMAMOTO, K. UND TACHIBANA, K. (2000): Virtual Object Manipulation on a Table-Top AR Environment, In: *Proceedings of the IEEE and ACM International Symposium on Augmented Reality (ISAR'00)*, S. 111–119.
- [KEAHEY 1998] KEAHEY, T.A. (1998): The Generalized Detail-In-Context Problem, In: *Proceedings of the 1998 IEEE Symposium on Information Visualization (InfoVis'98)*, IEEE Computer Society Press, S. 44–51.
- [KEEFE U. A. 2009] KEEFE, D., EWERT, M., RIBARSKY, W. UND CHANG, R. (2009): Interactive Coordinated Multiple-View Visualization of Biomechanical Motion Data, In: *IEEE Transactions on Visualization and Computer Graphics*, Vol.15/6, S. 1383–1390.
- [KELLER & KELLER 1993] KELLER, P.R. UND KELLER, M.M. (1993): *Visual Cues: Practical Data Visualization*, IEEE Computer Society Press.
- [KLEPPE U. A. 2003] KLEPPE, A.G., WARMER, J.B. UND BAST, W. (2003): *MDA Explained: The Model Driven Architecture: Practice and Promise*, Addison-Wesley.
- [KOSARA U. A. 2003] KOSARA, R., HEALEY, C.G., INTERRANTE, V., LAIDLAW, D.H. UND WARE, C. (2003): User Studies: Why, How, and When?, In: *IEEE Computer Graphics and Applications*, Vol.23/4, S. 20–25.
- [KUSCH U. A. 2010] KUSCH, K., FRANKE, I.S., HELMERT, J., WOJZIAK, J., MOSCH, M., PANNASCH, S., GROH, R. UND VELICHKOVSKY, B.M. (2010): Sakkadenkontingente perspektivische Optimierung von Bewegtbildern, In: *Tagungsband der Tagung experimentell arbeitender Psychologen (TeaP'10)*.
- [LAMAR U. A. 2001] LAMAR, E., HAMANN, B. UND JOY, K.I. (2001): A Magnification Lens for Interactive Volume Visualization, In: *Proceedings of the Ninth Pacific Conference on Computer Graphics and Applications (PG'01)*, IEEE Computer Society Press, S. 223–232.
- [LANGE U. A. 2006] LANGE, S., NOCKE, T. UND SCHUMANN, H. (2006): Visualisierungsdesign – ein systematischer Überblick, In: *Proceedings of Simulation and Visualization (SimVis'06)*, SCS Publishing House e.V., S. 113–128.
- [LEVENE 1998] LEVENE, J. (1998): A Framework for Non-Realistic Projections, Master Thesis, Massachusetts Institute of Technology, Cambridge, MA, USA.
- [LIMBOURG & VANDERDONCKT 2004] LIMBOURG, Q. UND VANDERDONCKT, J. (2004): USIXML: A User Interface Description Language Supporting Multiple Levels of Independence, In: *Proceedings of the International Conference on Web Engineering (ICWE'04)*, Rinton Press, S. 325–338.
- [LÖFFELMANN & GRÖLLER 1996] LÖFFELMANN, H. UND GRÖLLER, E. (1996): Ray Tracing with Extended Cameras, In: *The Journal of Visualization and Computer Animation*, Vol.7/4, S. 211–227.
- [LORAN 2006] LORAN, E. (2006): *Cézanne's Composition: Analysis of His Form with Diagrams and Photographs of His Motifs*, University of California Press.
- [LORENZ & DÖLLNER 2009] LORENZ, H. UND DÖLLNER, J. (2009): Real-time Piecewise Perspective Projections, In: *Proceedings of the International Conference on Computer Graphics Theory and Applications (GRAPP'09)*, INSTICC Press, S. 147–155.
- [LORENZ U. A. 2008] LORENZ, H., TRAPP, M., DÖLLNER, J. UND JOBST, M. (2008): Interactive Multi-Perspective Views of Virtual 3D Landscape and City Models, In: *Lecture Notes in Geoinformation and Cartography*. Springer-Verlag, S. 301–321.



- [LUFT U. A. 2006] LUFT, T., COLDITZ, C. UND DEUSSEN, O. (2006): Image Enhancement by Unsharp Masking the Depth Buffer, In: *ACM Transactions on Graphics*, Vol.25/3, S. 1206–1213.
- [LUFT U. A. 2008] LUFT, T., KOB, F., ZINSER, W. UND DEUSSEN, O. (2008): Watercolor Illustrations of CAD Data. In: *Proceedings of the Fourth Eurographics conference on Computational Aesthetics in Graphics, Visualization and Imaging (CAE'08)*, Eurographics Association, S. 57–63.
- [LUYTEN 2004] LUYTEN, K. (2004): *Dynamic User Interface Generation for Mobile and Embedded Systems with Model-Based User Interface Development*, Dissertation, Transnationale Universiteit Limburg, Niederlande/Belgien.
- [LUYTEN U. A. 2004] LUYTEN, K., ABRAMS, M., LIMBOURG, Q. UND VANDERDONCKT, J. (2004): Developing User Interfaces with XML: Advances on User Interface Description Languages, In: *Sattelite Workshop of Advanced Visual Interfaces (AVI'04)*.
- [MACHOVER 1994] MACHOVER, C. (1994): Four Decades of Computer Graphics, In: *IEEE Computer Graphics and Applications*, Vol.14/6, S. 14–19.
- [MACKINLAY 1986] MACKINLAY, J. (1986): Automating the Design of Graphical Presentations of Relational Information, In: *ACM Transactions on Graphics*, Vol.5/2, S. 110–141.
- [MACKINLAY U. A. 1990] MACKINLAY, J.D., CARD, S.K. UND ROBERTSON, G.G. (1990): Rapid Controlled Movement Through a Virtual 3D Workspace, In: *Proceedings of the 17th Annual Conference on Computer Graphics and Interactive Techniques*, Vol.24/4, S. 171–176.
- [MAHEMOFF & JOHNSTON 1998] MAHEMOFF, M.J. UND JOHNSTON, L.J. (1998): Pattern Languages for Usability: An Investigation of Alternative Approaches, In: *Proceedings of the Third Asian Pacific Computer and Human Interaction (APCHI'98)*, IEEE Computer Society Press, S. 25 –30.
- [MAPLE U. A. 2004] MAPLE, C., MANTON, R. UND JACOBS, H. (2004): The Use of Multiple Co-ordinated Views in Three-dimensional Virtual Environments, In: *Proceedings of the 8. International Conference on Information Visualisation (IV'04)*, IEEE Computer Society Press, S. 778–784.
- [MARIAGE U. A. 2004] MARIAGE, C., VANDERDONCKT, J. UND PRIBÉANU, C. (2004): State of the Art of Web Usability Guidelines, In: *The Handbook of Human Factors in Web Design*. CRC Press, S. 688–700.
- [MASHIO U. A. 2010] MASHIO, K., YOSHIDA, K., TAKAHASHI, S. UND OKADA, M. (2010): Automatic Blending of Multiple Perspective Views for Aesthetic Composition, In: *Smart Graphics*. Springer Berlin Heidelberg, S. 220–231.
- [MASSON 2007] MASSON, T. (2007): *CG101: A Computer Graphics Industry Reference*, Digital Fauxtography.
- [MCCLOUD 1999] MCCLOUD, S. (1999): *Comics richtig lesen*, Carlsen Verlag.
- [MCGUFFIN U. A. 2003] MCGUFFIN, M.J., TANCAU, L. UND BALAKRISHNAN, R. (2003): Using Deformations for Browsing Volumetric Data, In: *Proceedings of the 14th IEEE Visualization 2003 (VIS '03)*, IEEE Computer Society Press, S. 401–408.
- [MEIER 1996] MEIER, B.J. (1996): Painterly Rendering for Animation, In: *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH'96)*, ACM Press, S. 477–484.
- [MEMMEL 2009] MEMMEL, T. (2009): *User Interface Specification for Interactive Software Systems : Process-, Method- and Tool-Support for Interdisciplinary and Collaborative Requirements Modelling and Prototyping-Driven User Interface Specification*, Dissertation, Universität Konstanz, Konstanz.

- [MEMMEL U. A. 2008] MEMMEL, T., GEIS, T. UND REITERER, H. (2008): Methoden, Notationen und Werkzeuge zur Übersetzung von Anforderungen in User Interface Spezifikationen, In: *Usability professionals 2008: Berichtband des sechsten Workshops des German Chapters der Usability Professionals Association* e.V. Fraunhofer IRB Verlag, S. 45–48.
- [MEMMEL & REITERER 2008] MEMMEL, T. UND REITERER, H. (2008): User-Interface-Entwicklung mit interaktiven Spezifikationen, In: *Mensch & Computer 2008* (MuC'08), Oldenbourg Wissenschaftsverlag, S. 357–366.
- [METZKER & REITERER 2002] METZKER, E. UND REITERER, H. (2002): Evidence-Based Usability Engineering, In: *Proceedings of the 4th International Conference on Computer-Aided Design of UIs* (CADUI'02), Springer-Verlag, S. 323–336.
- [MITCHELL U. A. 2007] MITCHELL, J.L., FRANCKE, M. UND ENG, D. (2007): Illustrative Rendering in Team Fortress 2, In: *Proceedings of the 5th International Symposium on Non-Photorealistic Animation and Rendering* (NPAR'07), ACM Press, S. 71–76.
- [MULLET & SCHIANO 1995] MULLET, K. UND SCHIANO, D.J. (1995): 3D or not 3D: "more is better" or "less is more"?, In: *Proceedings of CHI'95 Conference on Human Factors in Computing Systems* (CHI'95), ACM Press, S. 174–175.
- [MÜNCH 2008] MÜNCH, T. (2008): *Entwicklung eines multiperspektivischen Abbildungssystems für die Arbeitsumgebung BiLL (Bildsprache LiveLab)*, Diplomarbeit, Technische Universität Dresden, Dresden, Deutschland.
- [MUNZNER 1997] MUNZNER, T. (1997): H3: Laying Out Large Directed Graphs in 3D Hyperbolic Space, In: *Proceedings of the IEEE Symposium on Information Visualization 1997* (InfoVis'97), IEEE Computer Society Press, S. 2–10.
- [MYERS U. A. 2000] MYERS, B., HUDSON, S.E. UND PAUSCH, R. (2000): Past, Present, and Future of User Interface Software Tools, In: *ACM Transaction on Computer-Human Interaction*, Vol.7/1, S. 3–28.
- [MYERS & ROSSON 1992] MYERS, B.A. UND ROSSON, M.B. (1992): Survey on User Interface Programming, In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '92), ACM, S. 195–202.
- [NAUJOKS 2008] NAUJOKS, M. (2008): XAML–WPF per XML, Studienarbeit, Hochschule Karlsruhe Technik & Wirtschaft Karlsruhe, Karlsruhe.
- [NIELSEN 1994] NIELSEN, J. (1994): *Usability Engineering*, Morgan Kaufmann.
- [NIELSEN 1998] NIELSEN, J. (1998): *2D is better than 3D*, Manuskript, Nielsen Norman Group, Fremont, CA, USA.
- [NOCKE & SCHUMANN 2004] NOCKE, T. UND SCHUMANN, H. (2004): Goals of Analysis for Visualization and Visual Data Mining Tasks, In: *CODATA Workshop Information, Presentation and Design*.
- [NONAKA U. A. 2011] NONAKA, H., MARSICO, G., DE BACK, W., KALAIIDZIDIS, Y. UND ZERIAL, M. (2011): Understanding hepatocyte polarity and lumen formation: lessons from development, In: *Proceedings of the First Virtual Liver Retreat*.
- [NORTH & SHNEIDERMAN 2000] NORTH, C. UND SHNEIDERMAN, B. (2000): Snap-Together Visualization: A User Interface for Coordinating Visualizations via Relational Schemata, In: *Proceedings of the working conference on Advanced Visual Interfaces* (AVI'00), ACM Press, S. 128–135.
- [NORTH & SHNEIDERMAN 1997] NORTH, C.L. UND SHNEIDERMAN, B. (1997): *A Taxonomy of Multiple Window Coordinations*, Technischer Bericht, Department of Computer Science, University of Maryland, USA.
- [OESTEREICH 2005] OESTEREICH, B. (2005): *Die UML 2.0 Kurzreferenz für die Praxis*, Oldenbourg Verlag.

- [OSGi ALLIANCE 2007] OSGi ALLIANCE. (2007): *OSGi Service Platform Release 4, Version 4.1*, Spezifikation, OSGi Alliance, San Ramon, CA, USA.
- [PANOFKY 1980] PANOFKY, E. (1980): Aufsätze zu Grundfragen der Kunstwissenschaft, In: *H. Oberer und E. Verheyen*, S. 169–204.
- [PASEWALDT U. A. 2011] PASEWALDT, S., TRAPP, M. UND DÖLLNER, J. (2011): Multiscale Visualization of 3D Geovirtual Environments Using View-Dependent Multi-Perspective Views., In: *Journal of WSCG*, Vol.19/3, S. 111–118.
- [PATERNO U. A. 2009] PATERNO, F., SANTORO, C. UND SPANO, L.D. (2009): MARIA: A Universal, Declarative, Multiple Abstraction-Level Language for Service-Oriented Applications in Ubiquitous Environments, In: *ACM Transaction on Computer-Human Interaction*, Vol.16/4, S. 19:1–19:30.
- [PETRASCH 2007] PETRASCH, R. (2007): Model Based User Interface Design: Model Driven Architecture und HCI Patterns, In: *Mitteilungen der Gesellschaft für Informatik*, Vol.27/3, S. 5–10.
- [PETRASCH & MEIMBERG 2006] PETRASCH, R. UND MEIMBERG, O. (2006): *Model-Driven Architecture: Eine praxisorientierte Einführung in die MDA*, dpunkt Verlag.
- [PIERCE 2001] PIERCE, J.S. (2001): *Expanding the Interaction Lexicon for 3D Graphics*, Dissertation, Carnegie Mellon University, Pittsburgh, PA, USA.
- [PINHEIRO DA SILVA & PATON 2003] PINHEIRO DA SILVA, P. UND PATON, N.W. (2003): Improving UML Support for User Interface Design: A Metric Assessment of UMLi, In: *Proceedings of the Workshop on Bridging the Gaps Between Software Engineering and Human-Computer Interaction at International Conference on Software Engineering (ICSE '03)*, S. 76–83.
- [PLUMLEE & WARE 2003A] PLUMLEE, M. UND WARE, C. (2003): Integrating Multiple 3D Views through Frame-of-Reference Interaction, In: *Proceedings of the Conference on Coordinated and Multiple Views In Exploratory Visualization (CMV'03)*, IEEE Computer Society Press, S. 34–43.
- [PLUMLEE & WARE 2003B] PLUMLEE, M. UND WARE, C. (2003): An Evaluation of Methods for Linking 3D Views, In: *Proceedings of the 2003 Symposium on Interactive 3D Graphics (I3D'03)*, ACM Press, S. 193–201.
- [POPESCU U. A. 2009] POPESCU, V., ROSEN, P. UND ADAMO-VILLANI, N. (2009): The Graph Camera, In: *ACM Transactions on Graphics*, Vol.28/5, S. 1–8.
- [PREIM & DACHSELT 2010] PREIM, B. UND DACHSELT, R. (2010): *Interaktive Systeme: Band 1: Grundlagen, Graphical User Interfaces, Informationsvisualisierung*, Springer-Verlag.
- [PUERTA 1997] PUERTA, A.R. (1997): A Model-Based Interface Development Environment, In: *IEEE Software*, Vol.14/4, S. 40–47.
- [QI U. A. 2006] QI, W., TAYLOR, R.M., HEALEY, C.G. UND MARTENS, J.-B. (2006): A Comparison of Immersive HMD, Fish Tank VR and Fish Tank with Haptics Displays for Volume Visualization, In: *Proceedings of the 3rd Symposium on Applied Perception in Graphics and Visualization (APGV'06)*, ACM Press, S. 51–58.
- [RADEMACHER 1999] RADEMACHER, P. (1999): View-Dependent Geometry, In: *Proceedings of the 26th annual conference on Computer graphics and interactive techniques (SIGGRAPH'99)*, ACM Press, S. 439–446.
- [RADEMACHER & BISHOP 1998] RADEMACHER, P. UND BISHOP, G. (1998): Multiple-Center-of-Projection Images, In: *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH'98)*, ACM Press, S. 199–206.
- [REINERS U. A. 2002] REINERS, D., VOSS, G. UND BEHR, J. (2002): OpenSG: Basic Concepts, In: *Proceedings of the 1. OpenSG Symposium*.

- [REKIMOTO & GREEN 1993] REKIMOTO, J. UND GREEN, M. (1993): The Information Cube: Using Transparency in 3D Information Visualization, In: *Proceedings of the Third Annual Workshop on Information Technologies & Systems (WITS'93)*, S. 125–132.
- [ROBERTS 2007] ROBERTS, J.C. (2007): State of the Art: Coordinated & Multiple Views in Exploratory Visualization, In: *Proceedings of the Fifth International Conference on Coordinated and Multiple Views in Exploratory Visualization (CMV'07)*, IEEE Computer Society Press, S. 61–71.
- [ROBERTSON U. A. 1993] ROBERTSON, G.G., CARD, S.K. UND MACKINLAY, J.D. (1993): Information Visualization using 3D Interactive Animation, In: *Communications of the ACM*, Vol.36/4, S. 57–71.
- [ROBERTSON U. A. 1991] ROBERTSON, G.G., MACKINLAY, J.D. UND CARD, S.K. (1991): Cone Trees: Animated 3D Visualizations of Hierarchical Information, In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI'91)*, ACM Press, S. 189–194.
- [ROBERTSON 1990] ROBERTSON, P.K. (1990): A Methodology for Scientific Data Visualisation: Choosing Representations Based on a Natural Scene Paradigm, In: *Proceedings of the First IEEE Conference on Visualization (Visualization'90)*, IEEE Computer Society Press, S. 114–123.
- [ROHLF & HELMAN 1994] ROHLF, J. UND HELMAN, J. (1994): IRIS Performer: A High Performance Multiprocessing Toolkit for Real-Time 3D Graphics, In: *Proceedings of the 21st Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '94)*, ACM Press, S. 381–394.
- [ROMAN U. A. 2004] ROMAN, A., GARG, G. UND LEVOY, M. (2004): Interactive Design of Multi-Perspective Images for Visualizing Urban Landscapes, In: *Proceedings of the Conference on Visualization (VIS'04)*, IEEE Computer Society Press, S. 537–544.
- [ROSEN & POPESCU 2011] ROSEN, P. UND POPESCU, V. (2011): An evaluation of 3-D scene exploration using a multiperspective image framework, In: *The Visual Computer*, Vol.27/6, S. 623–632.
- [ROSSON & CARROLL 2002] ROSSON, M.B. UND CARROLL, J.M. (2002): *Usability Engineering: Scenario-Based Development of Human-Computer Interaction*, Morgan Kaufmann.
- [ROTH & MATTIS 1990] ROTH, S.F. UND MATTIS, J. (1990): Data Characterization for Intelligent Graphics Presentation, In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems: Empowering People (CHI'90)*, ACM Press, S. 193–200.
- [SALOMON 2006] SALOMON, D. (2006): *Transformations and Projections in Computer Graphics*, Springer Verlag.
- [SCHIRRA 2006] SCHIRRA, J.R.J. (2006): Computational Visualistics: Dealing with Pictures in Computer Science, In: *Bildwissenschaft zwischen Reflexion und Anwendung*. Herbert von Halem Verlag, S. 494–509.
- [SCHIRRA & SCHOLZ 2000] SCHIRRA, J.R.J. UND SCHOLZ, M. (2000): Zwei Skizzen zum Begriff 'Photorealismus' in der Computergraphik, In: *Bild - Bildwahrnehmung - Bildverarbeitung: interdisziplinäre Beiträge zur Bildwissenschaft*. Deutscher Wissenschafts-Verlag, S. 69–79.
- [SCHLUNGBAUM & ELWERT 1996] SCHLUNGBAUM, E. UND ELWERT, T. (1996): Automatic User Interface Generation from Declarative Models, In: *Proceedings of the International Conference on Computer-Aided Design of User Interfaces (CADUI'96)*, University Press, S. 3–18.
- [SCHMIDT 2000] SCHMIDT, A. (2000): Implicit Human Computer Interaction Through Context, In: *Personal and Ubiquitous Computing*, Vol.4/2, S. 191–199.

- [SCHMIDT 2009] SCHMIDT, A. (2009): *Untersuchung der Darstellungsmöglichkeiten krummliniger Perspektiven*, Diplomarbeit, Technische Universität Dresden, Dresden, Deutschland.
- [SCHNEIDER 2008] SCHNEIDER, W. (2008): *Ergonomische Gestaltung von Benutzungsschnittstellen: Kommentar zur Grundsatznorm DIN EN ISO 9241-110*, Beuth.
- [SCHÖNHAGE 2001] SCHÖNHAGE, B. (2001): *DIVA Architectural Perspectives on Information Visualization*, Dissertation, Vrije Universiteit, Amsterdam.
- [SCHROEDER & LORENSON 1996] SCHROEDER, W. UND LORENSON, B. (1996): *Visualization Toolkit: An Object-Oriented Approach to 3-D Graphics*, Prentice Hall.
- [SCHUMANN & MÜLLER 2000] SCHUMANN, H. UND MÜLLER, W. (2000): *Visualisierung: Grundlagen und allgemeine Methoden*, Springer-Verlag.
- [SEITZ & KIM 2003] SEITZ, S.M. UND KIM, J. (2003): Multiperspective Imaging, In: *IEEE Computer Graphics and Applications*, Vol.23/6, S. 16–19.
- [SHAER U. A. 2008] SHAER, O., JACOB, R.J.K., GREEN, M. UND LUYTEN, K. (2008): User Interface Description Languages for Next Generation User Interfaces, In: *Extended Abstracts on Human Factors in Computing Systems (CHI'08)*, ACM Press, S. 3949–3952.
- [SHAMIR U. A. 2006] SHAMIR, A., RUBINSTEIN, M. UND LEVINBOIM, T. (2006): Generating Comics from 3D Interactive Computer Graphics, In: *IEEE Computer Graphics and Applications*, Vol.26/3, S. 53–61.
- [SHNEIDERMAN 1996] SHNEIDERMAN, B. (1996): The Eyes Have It: A Task by Data Type Taxonomy for Information Visualizations, In: *Proceedings of the IEEE Symposium on Visual Languages (VL'96)*, IEEE Computer Society Press, S. 336–343.
- [SHNEIDERMAN 2003] SHNEIDERMAN, B. (2003): Why not make interfaces better than 3D reality?, In: *IEEE Computer Graphics and Applications*, Vol.23/6, S. 12–15.
- [SHNEIDERMAN 2004] SHNEIDERMAN, B. (2004): *Designing the User Interface. Strategies for Effective Human-Computer Interaction*, Addison-Wesley.
- [SINGH 2002] SINGH, K. (2002): A Fresh Perspective, In: *Proceedings of Graphics Interface (GI'02)*, A. K. Peters Ltd., S. 17–24.
- [SINGH & BALAKRISHNAN 2004] SINGH, K. UND BALAKRISHNAN, R. (2004): Visualizing 3D Scenes using Non-Linear Projections and Data Mining of Previous Camera Movements, In: *Proceedings of the 3rd International Conference on Computer Graphics, Virtual Reality, Visualisation and Interaction in Africa (AFRIGRAPH'04)*, ACM Press, S. 41–48.
- [SMALLMAN U. A. 2001] SMALLMAN, H.S., JOHN, M.S., OONK, H.M. UND COWEN, M.B. (2001): Information Availability in 2D and 3D Displays, In: *IEEE Computer Graphics and Applications*, Vol.21/5, S. 51–57.
- [SMART U. A. 2006] SMART, J., HOCK, K. UND CSOMOR, S. (2006): *Cross-platform GUI programming with wxWidgets*, Prentice Hall.
- [SNYDER & VOXLAND 1989] SNYDER, J.P. UND VOXLAND, P.M. (1989): *An Album of Map Projections*, United States Government Printing.
- [SOUCHON & VANDERDONCKT 2003] SOUCHON, N. UND VANDERDONCKT, J. (2003): A Review of XML-compliant User Interface Description Languages, In: *Interactive Systems. Design, Specification, and Verification*. Springer-Verlag, S. 391–401.
- [SOUSA U. A. 2007] SOUSA, K., MENDONÇA, H. UND VANDERDONCKT, J. (2007): Towards Method Engineering of Model-Driven User Interface Development, In: *Proceedings of the 6th international conference on Task models and diagrams for user interface design (TAMODIA'07)*, Springer-Verlag, S. 112–125.



- [SOWIZRAL 2000] SOWIZRAL, H. (2000): Scene Graphs in the New Millennium, In: *IEEE Computer Graphics and Applications*, Vol.20/1, S. 56–57.
- [SPINDLER U. A. 2006] SPINDLER, M., BUBKE, M., GERMER, T. UND STROTHOTTE, T. (2006): Camera Textures, In: *Proceedings of the 4th International Conference on Computer Graphics and Interactive Techniques in Australasia and Southeast Asia (GRAPHITE'06)*, ACM Press, S. 295–302.
- [STAHL U. A. 2007] STAHL, T., VÖLTER, M., EFFTINGE, S. UND HAASE, A. (2007): *Modellgetriebene Softwareentwicklung: Techniken, Engineering, Management*, dpunkt Verlag.
- [STAPELKAMP 2007] STAPELKAMP, T. (2007): *Screen- und Interfacedesign. Gestaltung und Usability für Hard- und Software*, Springer-Verlag.
- [STARKE 2011] STARKE, M. (2011): *Interaktive Panelsetzung als Methode der strukturierten Betrachtung räumlicher Szenen in Echtzeitumgebungen*, Diplomarbeit, Technische Universität Dresden, Dresden, Deutschland.
- [STARKE U. A. 2011] STARKE, M., WOJDAK, J. UND GROH, R. (2011): Interactive Panels - A tool for structured three-dimensional scene exploration and visualization, In: *Workshop 3D-NordOst 2011 - Workshop zur Erfassung, Modellierung, Verarbeitung und Auswertung von 3D-Daten (3D-NordOst'11)*, Gesellschaft zur Förderung angewandter Informatik, S. 67–75.
- [STEED 2008] STEED, A. (2008): Some Useful Abstractions for Re-Usable Virtual Environment Platforms, In: *Proceedings of the IEEE Virtual Reality 2008 Workshop: Software Engineering and Architectures for Realtime Interactive Systems (SEARIS'08)*, Shaker Verlag, S. 33–36.
- [STOAKLEY U. A. 1995] STOAKLEY, R., CONWAY, M.J. UND PAUSCH, R. (1995): Virtual Reality on a WIM: Interactive Worlds in Miniature, In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI'95)*, ACM Press, S. 265–272.
- [STREITZ U. A. 2003] STREITZ, N., RÖCKER, C., PRANTE, T., STENZEL, R. UND VAN ALPHEN, D. (2003): Situated Interaction with Ambient Information: Facilitating Awareness and Communication in Ubiquitous Work Environments, In: *Hci International 2003 Proceedings: Human-Centered Computing: Cognitive, Social, and Ergonomic Aspects (HCII'03)*, CRC Press, S. 133–137.
- [STROTHOTTE U. A. 1994] STROTHOTTE, T., PREIM, B., RAAB, A., SCHUMANN, J. UND FORSEY, D.R. (1994): How to Render Frames and Influence People, In: *Computer Graphics Forum*, Vol.13/3, S. 455–466.
- [STROTHOTTE & SCHLECHTWEG 2002] STROTHOTTE, T. UND SCHLECHTWEG, S. (2002): *Non-Photorealistic Computer Graphics. Modeling, Rendering, and Animation*, Morgan Kaufmann.
- [STROUSTRUP 2000] STROUSTRUP, B. (2000): *Die C++-Programmiersprache*, Addison-Wesley.
- [STROUSTRUP 2007] STROUSTRUP, B. (2007): Evolving a language in and for the real world: C++ 1991–2006, In: *Proceedings of the third ACM SIGPLAN Conference on History of Programming Languages (HOPL'07)*, ACM Press, S. 4.1–4.59.
- [STUERZLINGER & WINGRAVE 2011] STUERZLINGER, W. UND WINGRAVE, C.A. (2011): The Value of Constraints for 3D User Interfaces, In: *Virtual Realities*. Springer-Verlag, S. 203–223.
- [SUDARSANAM U. A. 2008] SUDARSANAM, N., GRIMM, C. UND SINGH, K. (2008): Non-linear Perspective Widgets for Creating Multiple-View Images, In: *Proceedings of the 6th International Symposium on Non-Photorealistic Animation and Rendering (NPAR'08)*, ACM Press, S. 69–77.
- [SUTHERLAND 1964] SUTHERLAND, I.E. (1964): Sketch pad a man-machine graphical communication system, In: *Proceedings of the SHARE Design Automation Workshop (DAC'64)*, ACM Press, S. 6329–6346.

- [SZYPERSKI U. A. 1999] SZYPERSKI, C., BOSCH, J. UND WECK, W. (1999): Component-oriented programming, In: *Object-Oriented Technology ECOOP'99 Workshop Reader*. Springer-Verlag, S. 184–192.
- [TAKALA U. A. 2012] TAKALA, T.M., RAUHAMAA, P. UND TAKALA, T. (2012): Survey of 3DUI Applications and Development Challenges, In: *IEEE Symposium on 3D User Interfaces (3DUI'12)*, IEEE Computer Society Press, S. 89–96.
- [TESLER & STRASNICK 1992] TESLER, J. UND STRASNICK, S. (1992): *FSN: The 3D File System Navigator*, Dokumentation, Silicon Graphics Inc., Mountain View, CA, USA.
- [TIDWELL 2011] TIDWELL, J. (2011): *Designing interfaces*, O'Reilly Media.
- [TORY U. A. 2004] TORY, M., MOLLER, T., ATKINS, M.S. UND KIRKPATRICK, A.E. (2004): Combining 2D and 3D Views for Orientation and Relative Position Tasks, In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI'04)*, ACM Press, S. 73–80.
- [TORY & MÖLLER 2004] TORY, M. UND MÖLLER, T. (2004): Human Factors in Visualization Research, In: *IEEE Transactions on Visualization and Computer Graphics*, Vol.10/1, S. 72–84.
- [TUFTE 2001] TUFTE, E.R. (2001): *Visual Display of Quantitative Information*, Graphics Press.
- [VALLANCE & CALDER 2001A] VALLANCE, S. UND CALDER, P. (2001): Context in 3D Planar Navigation, In: *Proceedings of the Second Australasian User Interface Conference (AUIC'01)*, IEEE Computer Society Press, S. 93–99.
- [VALLANCE & CALDER 2001B] VALLANCE, S. UND CALDER, P. (2001): Multi-Perspective Images for Visualisation, In: *Proceedings of the Pan-Sydney Area Workshop on Visual Information Processing (VIP'01)*, Australian Computer Society, S. 69–76.
- [VANDERDONCKT U. A. 2004] VANDERDONCKT, J., LIMBOURG, Q., MICHOTTE, B., BOUILLON, L., TREVISAN, D. UND FLORINS, M. (2004): USIXML: a User Interface Description Language for Specifying Multimodal User Interfaces, In: *Proceedings of W3C Workshop on Multimodal Interaction (WMI'04)*, S. 1–7.
- [VITZTHUM 2008] VITZTHUM, A. (2008): *Entwicklungsunterstützung für interaktive 3D-Anwendungen*, Dissertation, LMU München, München, Deutschland.
- [WALSH 2002] WALSH, A.E. (2002): Understanding Scene Graphs, In: *Dr. Dobb's Journal*, Vol.27/7, S. 17–23.
- [WALTHER 1998] WALTHER, I.F. (1998): *Malerei der Welt*, Taschen Verlag.
- [WALTHER 2012] WALTHER, S. (2012): *Die umgekehrte Perspektive*, Diplomarbeit, Technische Universität Dresden, Dresden, Deutschland.
- [WANG BALDONADO U. A. 2000] WANG BALDONADO, M.Q., WOODRUFF, A. UND KUCHINSKY, A. (2000): Guidelines for Using Multiple Views in Information Visualization, In: *Proceedings of the Working Conference on Advanced Visual Interfaces (AVI'00)*, ACM Press, S. 110–119.
- [WANG U. A. 2005] WANG, L., ZHAO, Y., MUELLER, K. UND KAUFMAN, A. (2005): The Magic Volume Lens: An Interactive Focus+Context Technique for Volume Rendering, In: *Proceedings of the IEEE Visualization Conference (VIS'05)*, IEEE Computer Society Press, S. 367–374.
- [WANG & QIAN 2010] WANG, R. UND QIAN, X. (2010): *OpenSceneGraph 3.0: Beginner's Guide*, Packt Publishing.
- [WARE 2004] WARE, C. (2004): *nformation Visualization. Perception for Design.: Perception for Design*, Elsevier Ltd.
- [WARE 2009] WARE, W.R. (2009): *Modern Perspective: A Treatise Upon the Principles and Practice of Plane and Cylindrical Perspective*, Bibliobazaar.



- [WEHREND & LEWIS 1990] WEHREND, S. UND LEWIS, C. (1990): A Problem-Oriented Classification of Visualization Techniques, In: *Proceedings of the 1st Conference on Visualization (VIS'90)*, IEEE Computer Society Press, S. 139–143.
- [VAN WELIE & TRAETTEBERG 2000] VAN WELIE, M. UND TRAETTEBERG, H. (2000): Interaction Patterns in User Interfaces, In: *Proceedings of the 7th Pattern Languages of Programs Conference (PLoP'00)*, S. 13–16.
- [VAN WELIE & VAN DER VEER 2003] VAN WELIE, M. UND VAN DER VEER, G.C. (2003): Pattern Languages in Interaction Design: Structure and Organization, In: *Proceedings of Human-Computer-Interaction (INTERACT'03)*, IOS Press, S. 527–534.
- [WEN 1995] WEN, J. (1995): *Exploiting Orthogonality in Three Dimensional Graphics for Visualizing Abstract Data*, Technischer Bericht, Brown University, Providence, RI, USA.
- [WENZEL 2005] WENZEL, A. (2005): *Paul Cézanne : ein Leben für die Malerei*, Prestel.
- [WERNECKE 1994] WERNECKE, J. (1994): *The Inventor Mentor: Programming Object-Oriented 3D Graphics with Open Inventor, Release 2*, Addison-Wesley.
- [WIDJAJA U. A. 2009] WIDJAJA, Y.Y., PANG, C.N.I., LI, S.S., WILKINS, M.R. UND LAMBERT, T.D. (2009): The Interactorium: Visualising proteins, complexes and interaction networks in a virtual 3-D cell, In: *Proteomics*, Vol.9/23, S. 5309–5315.
- [WILLATS 1997] WILLATS, J. (1997): *Art and Representation: New Principles in the Analysis of Pictures*, Princeton University Press.
- [WINGRAVE 2008] WINGRAVE, C.A. (2008): *Concept-Oriented Design in Chasm: Conversational Domain Language Inspired 3D User Interface Design and Development*, Dissertation, Virginia Polytechnic Institute and State University, Blacksburg, USA.
- [WINGRAVE & BOWMAN 2008] WINGRAVE, C.A. UND BOWMAN, D.A. (2008): Tiered Developer-Centric Representations for 3D Interfaces: Concept-Oriented Design in Chasm, In: *Proceedings of the IEEE Virtual Reality Conference (VR'08)*, IEEE Computer Society Press, S. 193–200.
- [WINKENBACH & SALESIN 1994] WINKENBACH, G. UND SALESIN, D.H. (1994): Computer-Generated Pen-and-Ink Illustration, In: *Proceedings of the 21st Annual Conference on Computer graphics and Interactive Techniques (SIGGRAPH'94)*, ACM Press, S. 91–100.
- [WISS U. A. 1998] WISS, U., CARR, D. UND JONSSON, H. (1998): Evaluating Three-Dimensional Information Visualization Designs: a Case Study of Three Designs, In: *Proceedings of the IEEE Conference on Information Visualization (IV'98)*, IEEE Computer Society Press, S. 137–144.
- [WOJDZIAK 2007] WOJDZIAK, J. (2007): *Konzeption und Realisierung einer Komponentenarchitektur für die Arbeitsumgebung Bildsprache LiveLab (BiLL)*, Diplomarbeit, Technische Universität Dresden, Dresden, Deutschland.
- [WOJDZIAK U. A. 2011A] WOJDZIAK, J., KAMMER, D., FRANKE, I.S. UND GROH, R. (2011): BiLL: An Interactive Computer System for Visual Analytics, In: *Proceedings of the 3rd ACM SIGCHI Symposium on Engineering Interactive Computing Systems (EICS'11)*, ACM Press, S. 259–264.
- [WOJDZIAK U. A. 2011B] WOJDZIAK, J., KAMMER, D., ZAVESKY, M., MÜNCH, T., EBNER, T. UND FRANKE, I.S. (2011): Nichtlineare Abbildungssysteme in interaktiven Echtzeitumgebungen, In: *Wieder mehr Sehen! Aktuelle Einblicke in die Technische Visualistik*. TUDpress, S. 9–20.
- [WOJDZIAK U. A. 2011C] WOJDZIAK, J., STARKE, M. UND GROH, R. (2011): Coordinated Interaction for Enhanced Perception in Multiple Views, In: *Proceedings of the 8th Symposium on Applied Perception in Graphics and Visualization (APGV'11)*, ACM Press, S. 120.

- [WOJDZIAK U. A. 2011D] WOJDZIAK, J., ZAVESKY, M., FRANKE, I.S., LAMBECK, C. UND GROH, R. (2011): Guest Login – Visitor-Centred Information Design, In: *Proceedings of EVA London 2011: Electronic Visualisation & the Arts* (EVA'11), British Computer Society, S. 81–86.
- [WOJDZIAK U. A. 2011E] WOJDZIAK, J., ZAVESKY, M., KUSCH, K., WUTTIG, D., FRANKE, I.S. UND GROH, R. (2011): Figure Out Perspectives: Perceptually Realistic Avatar Visualization, In: *Proceedings of the IASTED Conference on Computer Graphics and Imaging* (CGIM'11), ACTA Press.
- [WOLFF 2011] WOLFF, A. (2011): *Modellbasierte Generierung von Benutzungsoberflächen*, Dissertation, Universität Rostock, Rostock.
- [WUTTIG 2008] WUTTIG, D. (2008): Konzeption und Realisierung von multiperspektivischen Projektionen durch mehrere Kameras in der Arbeitsumgebung Bildsprache LiveLab (BiLL), Belegarbeit, Technische Universität Dresden, Dresden, Deutschland.
- [YANG U. A. 2005] YANG, Y., CHEN, J.X. UND BEHESHTI, M. (2005): Nonlinear Perspective Projections and Magic Lenses: 3D View Deformation, In: *IEEE Computer Graphics and Applications*, Vol.25/1, S. 76–84.
- [YANKOVA & FRANKE 2008] YANKOVA, A. UND FRANKE, I.S. (2008): Angle of view vs. perspective distortion: a psychological evaluation of perspective projection for achieving perceptual realism in computer graphics, In: *Proceedings of the 5th Symposium on Applied Perception in Graphics and Visualization* (APGV'08), ACM Press, S. 204.
- [YI U. A. 2007] YI, J.S., YOUN, K., STASKO, J.T. UND JACKO, J.A. (2007): Toward a Deeper Understanding of the Role of Interaction in Information Visualization, In: *IEEE Transactions on Visualization and Computer Graphics*, Vol.13/6, S. 1224–1231.
- [YU & McMILLAN 2004] YU, J. UND McMILLAN, L. (2004): A Framework for Multiperspective Rendering, In: *Proceedings of Eurographics Symposium on Rendering* (EGSR'04), EUROGRAPHICS Association, S. 61–68, 408.
- [ZAVESKY 2011] ZAVESKY, M. (2011): Human models for studies on visual perception changes, In: *14. Anwendungsbezogener Workshop zur Erfassung, Modellierung, Verarbeitung und Auswertung von 3D-Daten* (3D-NordOst'11), Gesellschaft zur Förderung angewandter Informatik, S. 121–130.
- [ZAVESKY 2012] ZAVESKY, M. (2012): *Wahrnehmungsrealistische Projektion anthropomorpher Formen*, Dissertation, Technische Universität Dresden, Dresden, Deutschland.
- [ZAVESKY U. A. 2010] ZAVESKY, M., WOJDZIAK, J., FRANKE, I.S., LINKE, M., PETER, F. UND GROH, R. (2010): Mut zur Lücke – Zur Umsetzung von spezifischen Algorithmen für Designer, In: *4. Symposium Technisches Design – Kosten und Nutzen*, TUDpress, S. 234–249.
- [ZELNIK-MANOR U. A. 2005] ZELNIK-MANOR, L., PETERS, G. UND PERONA, P. (2005): Squaring the Circles in Panoramas, In: *Proceedings of the Tenth IEEE International Conference on Computer Vision* (ICCV'05), IEEE Computer Society Press, S. 1292–1299.
- [ZORIN & BARR 1995] ZORIN, D. UND BARR, A.H. (1995): Correction of Geometric Perceptual Distortions in Pictures, In: *Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques* (SIGGRAPH'95), ACM Press, S. 257–264.

# Verzeichnis eigener Publikationen

- FRANKE, I.S., WOJDZIAK, J. UND KAMMER, D. (2013): Gestenbasierte Steuerung in multiplen Sichten auf 3D-Szenarien. In: *Tagungsband der Fachtagung Digital Engineering der IFF-Wissenschaftstage 2013*, Springer Verlag, S. 213–220.
- KAMMER, D., HENKENS, D., WOJDZIAK, J. UND FREITAG, G. (2011): Formalization and Combination of Touch and Point Interaction. In: *Workshop on Engineering Patterns for Multitouch Interfaces (EICS'11)*, ACM Press.
- KAMMER, D., WOJDZIAK, J., EBNER, T. UND FRANKE, I.S. (2011): Bildsprache LiveLab: Ein komponentenorientiertes Framework im Bereich der experimentellen Computergrafik. In: *Wieder mehr Sehen! Aktuelle Einblicke in die Technische Visualistik*. TUDpress.
- KAMMER, D., MÜLLER, R., WOJDZIAK, J., ZAVESKY, M., WEYPRECHT, P., SCHÖNEFELD, F. UND GROH, R. (2012): Innovationspotenzial validieren: Anwendungsgebiete für die 3D-Plattform Bildsprache LiveLab, In: *Innovationsforum open4INNOVATION2012 regional kooperativ-global innovativ Beiträge zum Fachforum*, Technische Universität Dresden, S. 80–85.
- KAMMER, D., WOJDZIAK, J., EBNER, T. UND FRANKE, I.S. (2011): Bildsprache LiveLab: Ein komponentenorientiertes Framework im Bereich der experimentellen Computergrafik, In: *Wieder mehr Sehen! Aktuelle Einblicke in die Technische Visualistik*. TUDpress, S. 21–32.
- KAMMER, D., WOJDZIAK, J., EBNER, T., FRANKE, I.S. UND GROH, R. (2012): A component-oriented framework for experimental computer graphics. *Elsevier - Computer Standards & Interfaces*, Vol.34/1, S. 93–100.
- KAMMER, D., WOJDZIAK, J. UND GROH, R. (2013): Interactive Color Perspective for 3D Graphics Applications: Enhancing Depth Perception and the Understanding of Object Relations. In: *Proceedings of the HCI International 2013 (HCII'13)*.
- KAMMER, D., WOJDZIAK, J., KECK, M., TARANKO, S. UND GROH, R. (2010): Towards a Formalization of Multi-touch Gestures. In: *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces (ITS'10)*, ACM Press, S. 49–58.
- KECK, M., KAMMER, D., WOJDZIAK, J., TARANKO, S. UND GROH, R. (2010): DelViz: Untersuchen von Visualisierungsformen durch eine Klassifizierung beruhend auf Social Tagging. In: *GeNeMe - Generation Neue Medien*.
- KECK, M., KOALICK, G., KAMMER, D., TARANKO, S. UND WOJDZIAK, J. (2011): DelViz: Ein Werkzeug zur Exploration von Visualisierungen. In: *Wieder mehr Sehen! Aktuelle Einblicke in die Technische Visualistik*. TUDpress, Dresden, Deutschland.
- KUSCH, K., FRANKE, I.S., HELMERT, J., U. A. (2010): Sakkadenkontingente perspektivische Optimierung von Bewegtbildern. In: *Tagungsband der Tagung experimentell arbeitender Psychologen (TeaP'10)*.
- LAMBECK, C., WOJDZIAK, J. UND GROH, R. (2011): Facet Lens – local exploration and discovery in globally faceted data set. In: *Proceedings of the 2nd DESIRE Network Conference on Creativity and Innovation in Design*, ACM Press, S. 85–88.
- STARKE, M., WOJDZIAK, J. UND GROH, R. (2011): Interactive Panels - A tool for structured three-dimensional scene exploration and visualisation. In: *Workshop 3D-NordOst 2011 - Workshop zur Erfassung, Modellierung, Verarbeitung und Auswertung von 3D-Daten (3D-NordOst'11)*, Gesellschaft zur Förderung angewandter Informatik, S. 67–75.
- WOJDZIAK, J., KAMMER, D., FRANKE, I.S. UND GROH, R. (2011): BiLL: An Interactive Computer System for Visual Analytics. In: *Proceedings of the 3rd ACM SIGCHI Symposium on Engineering Interactive Computing Systems (EICS'11)*, ACM Press, S. 259–264.
- WOJDZIAK, J., KAMMER, D. UND GROH, R. (2013): Sequential Art in Real-time 3D Applications. In: *Proceedings of the HCI International 2013 (HCII'13)*.
- WOJDZIAK, J., KAMMER, D., MÜLLER, R., STAHL, A., ZAVESKY, M. UND GROH, R. (2013): Bildsprache LiveLab, In: *Informatik 2013*.
- WOJDZIAK, J., KAMMER, D., ZAVESKY, M., MÜNCH, T., EBNER, T. UND FRANKE, I.S. (2011): Nichtlineare Abbildungssysteme in interaktiven Echtzeitumgebungen. In: *Wieder mehr Sehen! Aktuelle Einblicke in die Technische Visualistik*. TUDpress, S. 9–20.
- WOJDZIAK, J., STARKE, M. UND GROH, R. (2011): Coordinated Interaction for Enhanced Perception in Multiple Views. In: *Proceedings of the 8th Symposium on Applied Perception in Graphics and Visualization (APGV'11)*, ACM Press, S. 120.

WOJDZIAK, J., ZAVESKY, M., FRANKE, I.S., LAMBECK, C. UND GROH, R. (2011): Guest Login – Visitor-Centred Information Design. In: *Proceedings of EVA London 2011: Electronic Visualisation & the Arts* (EVA'11), British Computer Society, S. 81–86.

WOJDZIAK, J., ZAVESKY, M., KUSCH, K., WUTTIG, D., FRANKE, I.S. UND GROH, R. (2011): Figure Out Perspectives: Perceptually Realistic Avatar Visualization. In: *Proceedings of the IASTED Conference on Computer Graphics and Imaging* (CGIM'11), ACTA Press.

ZAVESKY, M., WOJDZIAK, J., FRANKE, I.S., LINKE, M., PETER, F. UND GROH, R. (2010): Mut zur Lücke – Zur Umsetzung von spezifischen Algorithmen für Designer. In: *4. Symposium Technisches Design – Kosten und Nutzen*, TUDpress, S. 234–249.

ZAVESKY, M., WOJDZIAK, J., KUSCH, K., WUTTIG, D., FRANKE, I.S. UND GROH, R. (2011): Wahrnehmungsrealistische Darstellung von Avataren, In: *Wieder mehr Sehen! Aktuelle Einblicke in die Technische Visualistik*. TUDpress, S. 91–104.

ZAVESKY, M., WOJDZIAK, J., KUSCH, K., WUTTIG, D., FRANKE, I.S. UND GROH, R. (2011): An Individual Perspective - Perceptually Realistic Depiction Of Human Figures. In: *VISAPP 2011 Proceedings of the International Conference on Computer Vision Theory and Applications*, SciTePress – Science and Technology Publications, S. 313 – 319.